1 of 3
AD A

LEVEL III

PROCEEDINGS
of the

SECOND SEMINAR

ON THE

DOD COMPUTER SECURITY

INITIATIVE PROGRAM (2nd)

NATIONAL BUREAU OF STANDARDS
GAITHERSBURG, MARYLAND

DTIC
S ELECTE D
JUL 24 1981

D

JANUARY 15-17, 1980

AD A101992

DTIC FILE COPY

① 1780

① 261

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

81 7 22 119

CLEARED
FOR OPEN PUBLICATION

TABLE OF CONTENTS

JUL 1 3 1981   **13**   PAGE

DIRECTORATE FOR FREEDOM OF INFORMATION
AND SECURITY REVIEW (OASD-PA)
DEPARTMENT OF DEFENSE

i

TABLE OF CONTENTS (Continued)

PAGE

TABLE OF CONTENTS (Concluded)

iii

ABOUT THE SEMINAR

This is the second in a series of seminars to acquaint computer system
developers and users with the status of "trusted"* ADP system developments
within the Department of Defense and current planning for the integrity
evaluation of commercial implementations of similar systems. This seminar
will go into more detail both on the technical experiences of the DoD
research efforts in this area and the implications of trusted systems on
the use of computers.

Following the first day of topics of general interest the seminar will divide
into two parallel sessions. The technical session, intended for operating
system developers and sophisticated computer science technical experts,
will provide a detailed analysis of the Trusted Computing Base concept which
is the emerging generalized basis upon which high integrity operating systems
may be evaluated, followed by discussions by the principal designers of the
major DoD trusted system developments relating their systems to the Trusted
Computing Base Concept. The non-technical session will provide indepth
discussion of policy issues as they apply to multilevel secure computer
systems, an analysis of applications of such systems within the DoD and
beyond, and a not-so-technical review of the Trusted Computing Base concepts.

There will be extensive question and answer periods during both parallel
sessions and audience interaction is encouraged.

The Trusted Computing Base concept being introduced at this seminar is a
first draft specification against which the integrity of computer systems
may be evaluated. This draft specification is the result of much interaction
within the DoD community and is being introduced here to obtain reactions
from industry and other users. This draft specification is only a beginning
and is expected to change significantly as a result of interactions with
industry and government.

---

*A "trusted" ADP system is one which employs sufficient hardware and software
integrity measures to allow its use for simultaneously processing multiple
levels of classified and/or sensitive information.

ABOUT THE DoD COMPUTER SECURITY INITIATIVE

The Department of Defense (DoD) Computer Security Initiative was established
in 1978 by the Assistant Secretary of Defense for Communications, Command,
and Control and Intelligence to achieve the widespread availability of "trusted"
ADP systems for use within the DoD. Widespread availability implies the use
of commercially developed trusted ADP systems whenever possible. Recent DoD
research activities are demonstrating that trusted ADP systems can be developed
and successfully employed in sensitive information handling environments. In
addition to these demonstration systems, a technically sound and consistent
evaluation procedure must be established for determining the environments for
which a particular trusted system is suitable.

The Computer Security Initiative is attempting to foster the development
of trusted ADP systems through technology transfer efforts and to define
reasonable ADP system evaluation procedures to be applied to both government
and commercially developed trusted ADP systems. This seminar is the second
in a series which constitute an essential element in the Initiative's
Technology Transfer Program.

The Institute for Computer Sciences and Technology, through its Computer
Security and Risk Management Standards program, seeks new technology to
satisfy Federal ADP security requirements. The Institute then promulgates
acceptable and cost effective technology in Federal Information Processing
Standards and Guidelines. The Institute is pleased to assist the Department
of Defense in transferring the interim results of its research being conducted
under the Computer Security Initiative.

PROGRAM

Second Seminar on the Department of Defense Computer Security Initiative

National Bureau of Standards
Gaithersburg, Maryland

January 15, 1980   Red Auditorium

9:30 am   "The Impact of Computer Security in the Intelligence
                   Community"

Dr. John Koehler
Deputy Director for Central Intelligence for
      Resource Management

"The Impact of Computer Security in the Department
      of Defense"

Dr. Irwin Lebow
Chief Scientist
Defense Communications Agency

"The Impact of Computer Security in the Federal
      Government"

Mr. James Burrows
Director, Institute for Computer Science and
      Technology
National Bureau of Standards

BREAK

"The Impact of Computer Security in the Private
      Sector"

Mr. Ed Jacks
General Motors Corporation

"Status of the DoD Computer Security Initiative"

Mr. Stephen T. Walker
Chairman, DoD Computer Security Technical
      Consortium

1:00 pm   LUNCH

January 15, 1980
(Continued)

2:00 pm   "Computer Security Impacts on Near Term Systems"

Mr. Clark Weissman
System Development Corporation

"Computer Security Impacts on Future System
    Architectures"

Mr. Ed Burke
MITRE Corporation

BREAK

A "discussion" of what the computer manufacturers
would like/should expect to hear from government
users about trusted computer systems

Dr. Theodore M.P. Lee
UNIVAC Corporation

Mr. James P. Anderson
James P. Anderson Company

4:30 pm   ADJOURN

January 16-17, 1980   TWO PARALLEL SESSIONS

SESSION I   Gneral Session - Red Auditorium

January 16, 1980
9:15 am   "Policy Issues Relating to Computer Security"

Session Chairman: Robert Campbell
                         Advanced Information Management, Inc.

Mr. Cecil Phillips
Chairman, Computer Security Subcommittee
DCI Security Committee

Mr. Eugene Epperly
Counterintelligence & Security Policy Directorate
Office of the Secretary of Defense
Pentagon

Mr. Robert Campbell
Advanced Information Management, Inc.

Mr. Philip R. Manuel
Phillip R. Manuel and Associates

Dr. Stockton Gaines
RAND Corporation

1:00 pm   LUNCH

x

January 16, 1980
(Continued)

2:00 pm   "User Requirements and Applications"

Session Chairman: Dr. Stockton Gaines
RAND Corporation

Mr. Larry Bernosky
WWMCCS System Engineering Office

LtCol Cerny
Federal Republic of Germany Air Force

BREAK

Dr. Tom Berson
SYTEK Corporation

Mr. Mervyn Stuckey
U.S. Department of Housing and Urban Development

4:00 pm   ADJOURN

January 17, 1980

SESSION I

9:15 am   "User Requirements and Applications" (continued)

Dr. Von Der Brueck
IABG, Germany

Mr. John Rehbehn
Social Security Administration

Mr. William Nugent
Library of Congress

Mr. Howard Crumb
Federal Reserve Bank of New York

BREAK

"Trusted Computing Base Concepts"

Mr. Peter Tasker
MITRE Corporation

1:00 pm   LUNCH

2:00 pm   GENERAL DISCUSSION and WRAPUP

Mr. Stephen T. Walker

January 16 - 17   Green Auditorium

   SESSION II   Technical Session on Trusted Computing Base Design

This session, intended for operating system developers and sophisticated
computer science technical experts, will present the proposed Trusted
Computing Base (TCB) concept, an internal protection mechanism for high-
integrity, general-purpose computer systems. The most important issues
and tradeoffs in the design of a TCB for a general-purpose minicomputer
timesharing system will be discussed in detail. A technical background
in computer science will be assumed.

The first day will consist of a series of presentations by Mr. John P.L.
Woodward and Ms. Grace H. Nibaldi from the MITRE Corporation. First, the
concept of a Trusted Computing Base will be defined in detail, and the two
categories of TCB functions, software interface functions and user interface
functions will be discussed. Then each of these functional areas will be
discussed individually, with examples drawn from past and present TCB
developments.

At the end of the presentations, members of the audience will be asked to
identify design decisions they have faced in an operating system design or
similar effort to determine how these decisions relate to the choices that
must be made in TCB design. The audience will be encouraged to write their
thoughts down for possible discussion on the second day.

The second day will consist of a panel discussion by the developers of
the following trusted operation systems: ·

     Kernelized Secure Operating System (KSOS-11)
        Dr. McCauley, Ford Aerospace and Computer Corporation

     Kernelized Secure Operating System (KSOS-6)
        Mr. Bonneau, Honeywell Corporation

     UCLA Data Secure UNIX
        Dr. Popek, University of California at Los Angeles

     Kernelized VM-370 (KVM)
        Mr. Schaefer, System Development Corporation

The developers will describe their TCB's and discuss the design issues they
encountered and the tradeoffs they made in their designs. Following this, a
question, answer, and general discussion session will be held.

Extensive audience comments and questions are expected and encouraged.

Welcome and Opening Words

Stephen T. Walker

Chairman, DoD Computer Security
Technical Consortium

     Welcome to NBS and the Second Seminar on the DoD Computer Security
Initiative.

     My name is Stephen T. Walker and I am Chairman of the Computer
Security Technical Consortium which is responsible for the activities of
the Initiative.

     The goal of the Initiative is to achieve widespread availability
of the trusted computer systems.

     As your handout indicates, we define a "trusted" system as one with
sufficient hardware and software integrity measures to allow simultaneous
processing of multiple levels of sensitive information.

     That is a rather complicated definition that simply means we can rely
on the computer itself to protect information from unauthorized use or
modification or even more simply stated to make the computer work the way
we want it to.

     By widespread we meant generally available to a large customer base,
not special purpose.

     This is a complex problem, a subset of the overall computer security
problem (which includes physical, administration, personnel security)
one which hasn't received much attention until recently, largely because
it was felt by many to be too difficult to solve.

     Trusted computers are the subject of this seminar but before we go any
further, it is important to note that a solution to this problem does not
diminish the need for physical and administration security measures though
it may in some cases ease these measures for remote users at least.

     Many computer users especially those outside of the Department of
Defense have said (and undoubtedly will continue to say) our physical
security measures are so lax that we couldn't use a trusted computer if we
had one. Many have used this situation as an excuse not to worry about the
integrity inside their computers.

     Indeed, many facilities do not have reasonable physical and administrative
measures today. But there is a very important point here which must be
understood. If a computer user today wishes to take appropriate physical and
administrative security measures to protect his system (as many already do),
the technology needed to do this is readily available and well understood.
There are plenty of places to go for advice and help.

A-1

If, however, a user wishes to install a computer in which he does not have to treat all information and all users at the same sensitivity level, he doesn't have many options today. We are trying to change that situation.

I maintain that any computer facility processing information that is of any value at all, be it a large central system or a network of small systems or something in between, will eventually come up against the need for integrity within the computer system itself. We in the Department of Defense have been hampered by the lack of internal integrity within computers for some time, as many of the upcoming speakers will testify. But others, dealing with sensitive information outside of the national security classified world have also encountered this problem, and many of them will speak in the next three days.

One major premise, then, of the computer security initiative is that there is a widespread and growing need for computers with high levels of integrity within the Department of Defense, within the Federal Government, and within the private sector. The first major objective of the seminar is to make that widespread need very clear.

We hope this portion of the seminar will motivate the computer manufacturers to get involved in building high integrity systems to satisfy the full spectrum of needs of their customer base.

The solution to this system integrity problem will be difficult and will take time to accomplish on a broad scale. We in the DoD have been working on technical solutions for some years and we believe that some early solutions to the problem are now becoming available.

The techniques needed to develop a trusted computer system involve a strict adherence to good system development practices plus a strong dose of formal specification and verification both of the design and the implementation of a system. The stronger the successful dose of this specification and verification process the greater the confidence that can be placed on the system. These techniques are derived from and closely linked to the system development techniques evolving in the major computer science research centers around the country. We are not looking for radical changes in the state-of-the-art in system development. Unfortunately, most systems currently in development or being marketed by the manufacturers do not represent anything close to the state-of-the-art in system development and therefore a major shift will be required by many of the manufacturers, in effect catching up with the state-of-the-art, before trusted computers will become generally available.

A second major premise of the computer security initiative then is that the technology needed to develop trusted computer systems exists today.

The second major objective of this seminar is to describe in detail the experiences we have had in developing these systems. We hope this portion of the seminar will make clear what we feel are important steps needed to build high integrity systems and therefore make it easier for the manufacturers to develop such systems.

In reviewing the preregistered attendee list, I was again impressed, as with the first seminar which we held last July, with the broad spectrum of interests represented in the audience. I will give you a breakdown of who you are a little later, but I was reminded, as I thought of all the different perspectives here, of the familiar story which bears repeating, of the blind men who encountered an elephant for the first time and were asked to describe what an elephant is, based on their limited experiences. One, who encountered the elephant's leg insisted that an elephant is a tree because it is round and tall and had a rough skin. Another who encountered the flank insisted that an elephant must be a wall because it was tall and wide and flat. A third, who ran into the tail, insisted it must be a rope with a tassel on the end, while the fourth, feeling the trunk, was certain that an elephant must be a hose with the two holes in the end. We've all heard this story before but I feel it is particularly appropriate in this context because of the wide variety of backgrounds which we all represent. It is important as we begin these three days that each one of us realize that whatever our own perspective on the computer security problem, we are limited in our understanding by the experiences we have had. As we listen to the speakers, we should try hard to understand the perspective of the speaker and in that way broaden our understanding of the total computer security problem.

As indicated in Dr. Dineen's keynote address at the July Seminar, printed in your program, the Department of Defense views the problem of achieving a high degree of integrity in computer systems as very important to our future information handling needs. The DoD has in the past and will continue in the future to build special purpose systems to satisfy specific vital DoD needs, but our need for trusted computer systems goes well beyond our ability to build specialized DoD systems. Furthermore many of the reasons we need trusted systems, for protecting sensitive information be it classified data, personnel files, financial or logistic records, are not at all unlike the needs of the rest of the government and the private sector. We feel the best way to solve our needs in this area is to encourage the computer manufacturers to develop trusted systems so that all of us -- DoD, government and private sector users -- can make full and effective use of the results. In this way all our needs can be satisfied in the most general manner for the least expense to any of us.

Just getting the manufacturers to develop trusted systems isn't quite enough though. We will talk later in the seminar about the evolution of a process for evaluating the integrity of computer systems to determine the environments and applications for which they are suitable. What we will have to say about this is necessarily preliminary and not formalized in any sense. We realize that having some form of evaluation process readily available is essential to the acceptability of trusted computer systems and I can assure you we are hard at work trying to establish the proper and consistent procedures for evaluating computer systems in a way that will benefit all interested parties.

With all this as background then, let us begin.

A-3

"The Impact of Computer Security in
the Intelligence Community"

Dr. John Koehler
Deputy Director for Central Intelligence
for Resource Management

Outline of talk given at DoD Computer Security Conference –
January 15, 1980.

1. Introductory Remarks:

    a. Greetings from DCI

    b. Congratulations and thanks to sponsors – DoD and NBS

    c. DCI staff and all elements of the Intelligence Community
       cooperating with DoD Computer Security Initiative because
       of its importance to future progress in intelligence

2. The fundamental importance of computers

    a. The business of intelligence is information processing

    b. Time critical nature of intelligence information

    c. Time constraints, especially in I&W and for support to
       military operations have become progressively narrower

    d. Increasing demands on the Intelligence Community are
       apparent

       * The international environment is increasingly complex.
         We have spent great efforts to support treaty negotiations
         and monitoring; concurrently, the conflict and threat of
         conflict in peripheral areas require greater attention.

       * As with Defense, real resources going to the Intelligence
         Community have generally declined over the last decade.

       * Faced with this situation, there would have been no way
         in which the Intelligence Community could have continued
         to have fulfilled the expanding requirements levied upon
         it without a heavy reliance on increasingly complex and
         competent data processing and communications systems.

3. Computers, because of the historical pattern of their development,
   while helping to solve the Intelligence Community's problems
   have posed substantial new problems of their own--

    * The Intelligence Community always faces a dilemma: *because
      our sources are fragile, the information needs to be
      closely held, locked up, protected. But the purpose of
      gathering and producing intelligence is to help make
      better decisions. That requires data to be processed
      quickly and information disseminated quickly and broadly.*

    * The historical pattern of development of computer systems
      has made choosing among the two objectives of security and
      dissemination especially difficult.

    * The Von Neumann concept of the digital computer, implementing
      the computer's internal executive functions through programs
      executed in the same manner as applications programs,
      has made the present day computer highly vulnerable to
      sophisticated attack.

    * The relatively unstructured development of today's extremely
      large and complex operating systems has further exacerbated
      the problem.

    * The vulnerability which this has led to in today's computer
      systems, coupled with the concentration in one place of
      large masses of highly sensitive classified information,
      have posed problems with which it has proved extremely
      difficult to cope and which steadily increase in magnitude.

4. This is not to say that the Intelligence Community's present day
   information processing and handling systems pose an undue security
   hazard.

    * Faced with a lack of trusted software with which to implement
      security controls, we have had to rely heavily on physical
      security for systems to which all access is denied except
      to those cleared to the highest level of classification of
      any of the material present in any particular system.

    * To enforce such a security policy has, however, required
      considerable expensive duplication of resources, has placed
      a severe strain on security clearance procedures, and has
      restricted access to information more than we desire.

    * *As we strive to improve the efficiency with which the Intelligence
      Community executes its mission by taking advantage of the
      concept of distributed data bases and interactive real-time
      access to the expanding volumes of information which new
      techniques of internetting and high speed bulk transfer of
      data make possible, we can foresee the time when present
      solutions to the security problem will no longer be adequate.*

5. For the foregoing reasons, the DCI and his staff elements immediately concerned with the problem have taken an active part in supporting and encouraging the DoD Computer Security Initiative Program.

6. As Dr. Dineen noted in his keynote address to the first of these seminars, "The DoD cannot afford, just for the sake of having trusted computer systems, to develop its own general purpose hardware and software systems."

   * If the DoD, with all its resources cannot afford to do so, it goes without saying that the Intelligence Community cannot afford to do so either.

   * In the infancy of computer technology, because of the unavailability of commercial systems and the importance of the capabilities of the electronic computer to the Intelligence business, the Intelligence Community did just that.

   * Now, however, the magnitude, complexity and cost of today's generalized systems and their widespread availability in the commercial market militates against the development by the Intelligence Community of any but the most specialized intelligence processing systems.

   * Therefore, we, along with the DoD, must rely on and encourage the development of trusted systems by the industry.

   * To the extent that industry does meet the challenge of the security problem and produce systems which are demonstrably more secure, the Intelligence Community will provide a ready market for such products.

7. We recognize that as significant a portion of the intelligence and defense budgets as data processing and telecommunications are, the size of this market alone cannot totally justify the expenditure of the resources required to develop and market trusted software systems.

   * However, we are convinced that the problems of computer security are in no way limited to the Intelligence Community or the Department of Defense alone. They happen to be more critical in our areas of concern and therefore we are the most cognizant of them.

* As computers assume more and more decision making functions
and as more and more financial transactions are initiated
and executed by computers without human intervention,
making obsolete traditional control mechanisms and audit
functions, the problems of computer security will gain
greater and greater importance and attention in government
in all its functions and at all levels, national, state,
and local. Nor will business, particularly the financial
community, lag far behind this growing trend.

8. Thus, it is to the great mutual self-interest of ourselves and
the data processing industry that we exert the utmost effort to
work together to build on progress which has been made in this
area by the DoD Computer Security initiative.

* Just as the Intelligence Community cooperated with industry
and other elements of the Federal government in developing
the Data Encryption Standard in the field of communications
security, the Intelligence Community encourages and stands
ready to assist in any way that it can in the development
of improved computer security in the commercial field.

"THE IMPACT OF COMPUTER SECURITY IN THE DEPARTMENT
OF DEFENSE"

OUTLINE

- **THE DOD COMPUTER SECURITY CONTEXT**
    - • **NATIONAL LEVEL COMMAND AND CONTROL**
    - • **DCA RESPONSIBILITIES**
- **DEFINITION OF THE COMPUTER SECURITY PROBLEM**
- **OPERATIONAL LIMITATIONS RESULTING FROM CURRENT APPROACHES TO SECURITY**
- **EVOLUTION OF WWMCCS COMPUTER CONNECTIVITY**
- **LESSONS LEARNED**
- **CONCLUSIONS**

Dr. Irwin Lebow
Chief Scientist
Defense Communications Agency

MAJOR DCA RESPONSIBILITIES

SUBJECT	DCA RESPONSIBILITY
● WORLD-WIDE MILITARY COMMAND & CONTROL SYSTEM (WWMCCS)	ARCHITECTURE & SYSTEMS ENGINEERING
•• NATIONAL MILITARY COMMAND SYSTEM (NMCS)	DETAILED SYSTEMS ENGINEERING AND ADP SUPPORT
•• STANDARD ADP SYSTEMS	ARCHITECTURE, SYSTEMS ENGINEERING & COMMON SUPPORT
•• MINIMUM ESSENTIAL EMERGENCY COMMUNICATIONS NETWORK (MEECN)	SYSTEMS ENGINEERING
● DEFENSE COMMUNICATIONS SYSTEM	MANAGEMENT, ARCHITECTURE, SYSTEMS ENGINEERING
● MILITARY SATELLITE COMMUNICATIONS SYSTEMS	ARCHITECTURE

C-1

WWMCCS STANDARD ADP SYSTEMS

- CURRENT SYSTEM

  35 SITES (CONUS, PACIFIC, EUROPE)

  STANDARD HARDWARE

  - H6000 MAINFRAME
  - DN355 COMMUNICATIONS CONTROLLER
  - VISUAL INFO. PROJECTION TERMINALS

  STANDARD SOFTWARE

  - GCOS SYSTEM SOFTWARE
  - CERTAIN APPLICATIONS SOFTWARE

  NETWORKING WWMCCS COMPUTERS

  - CURRENTLY VIA DEDICATED WWMCCS INTERCOMPUTER NETWORK (WIN)
  - POST 1981 VIA COMMON USER AUTODIN II

- DEFINITION OF ALTERNATIVES FOR MAJOR UPGRADE NOW UNDERWAY

EVOLUTION OF DCS DATA SYSTEM

- CURRENTLY AUTODIN I

- 1980 - 1985 AUTODIN I/AUTODIN II

  - •• AUTODIN II PACKET SWITCHING BACKBONE

  - •• AUTODIN I MESSAGE SERVICE "HOSTS"

- POST 1985 AUTODIN II

  - •• AUTODIN I "HOSTS" PHASED OUT

  - •• MESSAGE SERVICE (AND OTHER SERVICES) PROVIDED ELSEWHERE

OPPOSING FORCES IN
THE EVOLUTION OF THE DCS

- DESIRE FOR MORE EFFECTIVE, EXTENSIVE
  INFORMATION SHARING AND EXCHANGE

- FEAR OF COMPROMISE OF SENSITIVE DATA

THE COMPUTER SECURITY PROBLEM

**TO PROTECT USER INFORMATION
ON SHARED COMPUTER SYSTEMS**

C-3

"PROTECT USER INFORMATION"

- PREVENT DATA COMPROMISE
- PREVENT DENIAL OF SERVICE
- GUARANTEE DATA INTEGRITY

"SHARED COMPUTER SYSTEMS"

- HOSTS
- HOST FRONT END DEVICES
- TEXT MESSAGE HANDLING SYSTEMS
- TERMINAL ACCESS CONTROLLERS
- SWITCHING NODES
- GATEWAYS ETC

ASPECTS OF THE
COMPUTER SECURITY PROBLEM

- AUTHENTICATION PROBLEM:

    PREVENTING PENETRATION BY UNAUTHORIZED USERS

- MULTILEVEL SECURITY PROBLEM:

    PREVENTING MISUSE BY AUTHORIZED USERS

APPROACHES TO AUTHENTICATION

IDENTIFY A USER BY

- WHAT HE IS

- WHAT HE KNOWS

- WHAT HE HAS

APPROACHES TO MULTILEVEL SECURITY

- LOGICALLY SEPARATE USERS: KERNEL TECHNOLOGY
- DISGUISE DATA: END-TO-END ENCRYPTION

WHY IS COMPUTER SECURITY A PROBLEM ?

- SENSITIVE/CLASSIFIED DATA STORED ONLINE

- MULTILEVEL SECURITY IS A DIFFICULT TECHNICAL PROBLEM

- SECURITY DEMANDS ARE INCREASING

- CURRENT APPROACHES ARE EXPENSIVE, INADEQUATE

CURRENT APPROACHES STRESS
PHYSICAL CONTROL

- PHYSICAL CONTROL OF ACCESS AREAS

- DEDICATED (REPLICATED) COMPUTER SYSTEMS

- PERIODS PROCESSING

- SYSTEM HIGH CLEARANCES

SECURITY PROBLEM AT HQ
FORCES COMMAND (FORSCOM)

H6000
DN 355

MUST SUPPORT CONNECTION TO
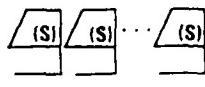
/(S)/ /(S)/ ··· /(S)/   JCS, USREDCOM, DA, LANTCOM, ...

>40 SECRET TERMINALS   WWMCCS COMPUTERS OPERATING
THROUGHOUT CONUS   AT TOP SECRET

C-7

FORSCOM SOLUTION: UPGRADE SOME TERMINALS, PERIODS PROCESS

SECRET SYSTEM

COLOR CHANGE

TOP SECRET SYSTEM

H6000 (S)

DN 355

(S) ··· (S)

SECRET TERMINALS WHICH CANNOT
BE CLEARED TO TOP SECRET

H6000 (TS)

DN 355

(TS) ··· (TS) JCS, USREDCOM, . . .

TERMINALS UPGRADED
TO TOP SECRET

TOP SECRET
WWMCCS
COMPUITERS

TIME
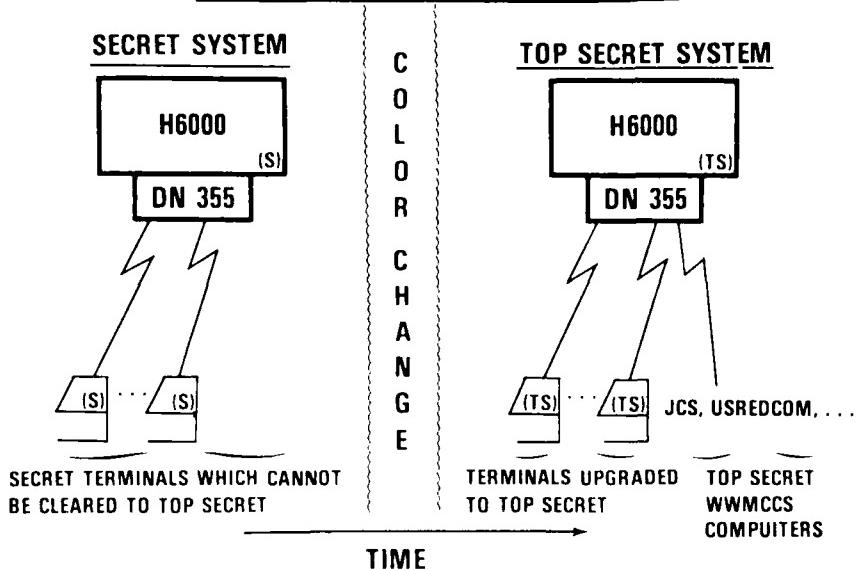
DRAWBACKS OF FORSCOM SOLUTION

- COST TO UPGRADE SECRET FACILITIES TO TOP SECRET

  - • EXTENSIVE PERSONNEL BACKGROUND INVESTIGATIONS

  - • MORE CUMBERSOME ADMINISTRATIVE, OPERATING PROCEDURES

- OPERATIONAL LIMITATIONS

  - • NO SERVICE DURING COLOR CHANGES

  - • INTERCONNECTIVITY RESTRICTED TO SPECIFIC PERIODS

SECURITY PROBLEM AT MILITARY
AIRLIFT COMMAND (MAC)

```
┌─────────────┐
│    H6000    │
└─────────────┘
  │ DN 355 │
```

MUST SUPPORT CONNECTION TO

(U) (U) ⋯ (U)	(TS) (TS) ⋯ (TS)	JCS, USREDCOM, DA, ...
TERMINALS SUPPORTING UNCLASSIFIED MISSIONS	TERMINALS SUPPORTING TOP SECRET MISSIONS	WWMCCS COMPUTERS OPERATING AT TOP SECRET
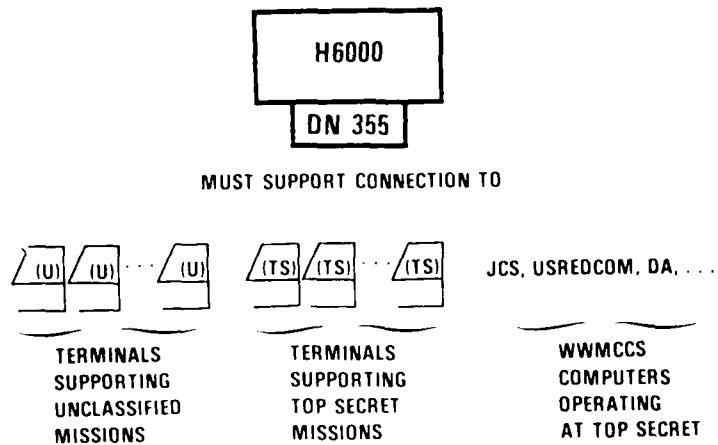
MAC SOLUTION: REPLICATE SYSTEMS

UNCLASSIFIED SYSTEM

```
┌─────────────┐
│    H6000    │
│        (U)  │
└─────────────┘
  │ DN 355 │
```

(U) ⋯ (U)

TOP SECRET SYSTEM

```
┌─────────────┐
│    H6000    │
│       (TS)  │
└─────────────┘
  │ DN 355 │
```

(TS) ⋯ (TS)   JCS, USREDCOM, ...

C-9
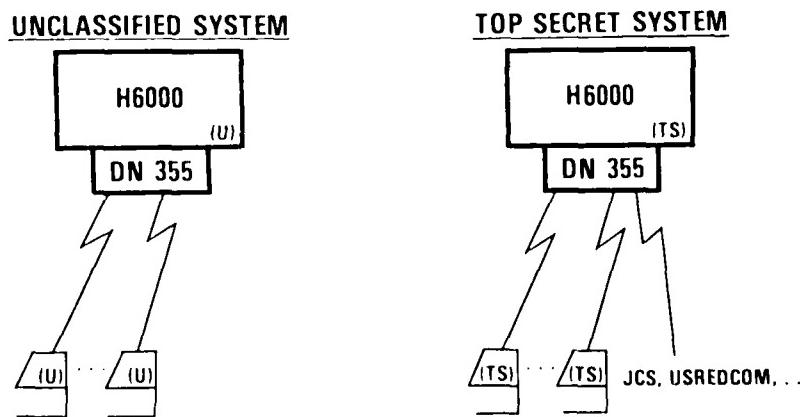
DRAWBACKS OF MAC SOLUTION

- COST TO REPLICATE MAINFRAMES
    - • REPLICATED PURCHASE, MAINTENANCE COST
    - • REPLICATED OPERATING EXPENSES

- OPERATIONAL LIMITATIONS
    - • TOP SECRET SYSTEM MUST ACCESS
      UNCLASSIFIED DATA BASES MANUALLY
      PROCEDURE INCONVENIENT, DATA NOT
      CURRENT
    - • COMPUTER RESOURCE CANNOT BE ASSIGNED
      DYNAMICALLY

EVOLUTION OF WWMCCS
COMPUTER CONNECTIVITY

ISOLATED SITES → ISOLATED SITES WITH REMOTE TERMINALS → SITES CONNECTED BY AUTODIN I → SITES CONNECTED BY AUTODIN II

SITES CONNECTED BY WIN

TIME ————————————————→
SECURITY RISK ————————————————→

C-10
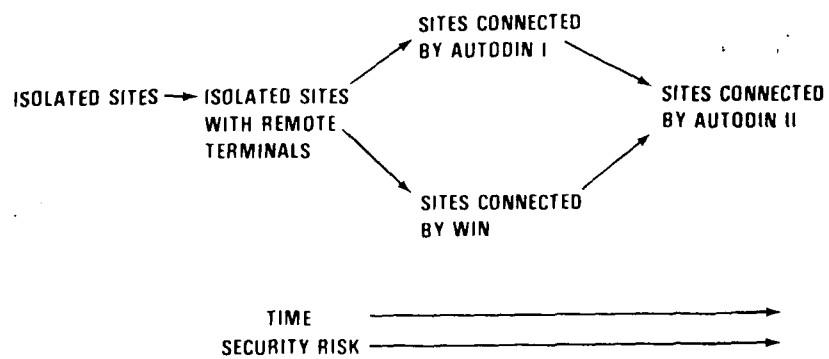
COMPUTER SECURITY
RISK IS A FUNCTION OF:

- COMPUTING ENVIRONMENT
    (SPAN OF PHYSICAL/ADMINISTRATIVE CONTROL)

- DEGREE OF USER FREEDOM
    (SPAN OF LOGICAL CONTROL)

TWO DIMENSIONS OF SECURITY RISK

HIGH

INCREASING SECURITY RISK

X SITES CONNECTED BY AUTODIN II

X SITES CONNECTED BY WIN

DEGREE OF USER FREEDOM

X SITES CONNECTED BY AUTODIN I

X ISOLATED SITES

X ISOLATED SITES WITH REMOTE TERMINALS

INCREASING SECURITY RISK

LOW

LOW   HIGH

DIFFICULTY OF PHYSICAL/ADMINISTRATIVE CONTROL
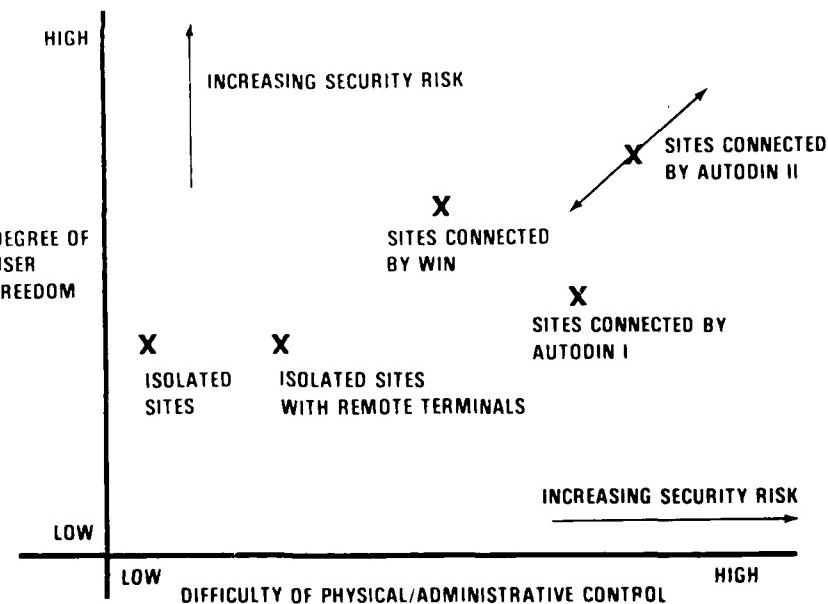
ISOLATED WWMCCS SITES
WITH LOCAL TERMINALS

- ○ "SHARING" AMONG LOCAL USERS
- ○ PHYSICAL/ADMINISTRATIVE CONTROL COMPLETE
- ◑ REPLICATED SYSTEMS
- ◕ SYSTEM HIGH CLEARANCES

```
┌─────────────────────┐
│      H6000           │
│   ┌─────────────┐    │
└───┤   DN 355    ├────┘
    └──┬───┬───┬──┘
       │   │   │
       T   T   T
```
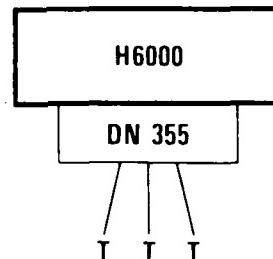
ISOLATED WWMCCS SITES
WITH REMOTE TERMINALS

- ◑ "SHARING" AMONG LOCAL USERS, REMOTE USERS ON
     DEDICATED LINES
- ○ LINK ENCRYPTION ON COMM LINES
- ◑ PHYSICAL/ADMINISTRATIVE CONTROL DISPERSED
- ◕ REPLICATED SYSTEMS
- ◕ SYSTEM HIGH CLEARANCES

```
┌─────────────────────┐
│      H6000           │              T
│   ┌─────────────┐    │          ╱
└───┤   DN 355    ├────┘───╱╲────╱
    └──┬───┬───┬──┘
       │   │   │
       T   T   T
```

C-12
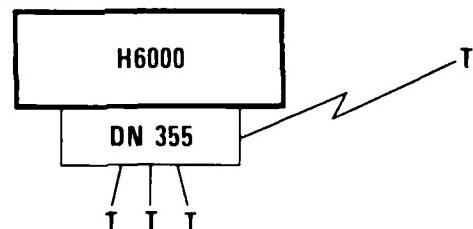
AUTODIN I

- INITIAL OPERATION 1963
- MESSAGE SWITCHING STORE-AND-FORWARD NETWORK
  - •• 9 U. S., 9 OVERSEAS SWITCHES
  - •• $\leq$ 9600 BAUD TRUNKS
- 2250 SUBSCRIBER TERMINATIONS
  - •• MILITARY AGENCIES
  - •• INTELLIGENCE AGENCIES
  - •• CIVILIAN AGENCIES
  - •• NO DIAL-UP ACCESS
- PROVIDES WRITTEN RECORD COMMUNICATIONS SERVICE
  - •• BATCH-STYLE SERVICE
  - •• FORMAL MESSAGE FORMATS

U.S. AUTODIN I

C-13
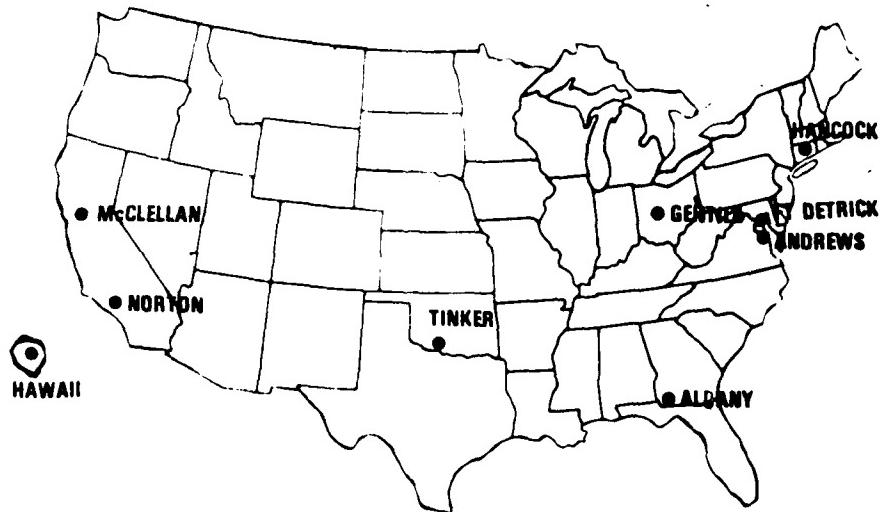
AUTODIN I SECURITY

- MULTILEVEL SECURE BY FIAT
    - ●● PHYSICAL SEPARATION OF USERS (R/Y)
    - ●● REDUNDANT SOFTWARE CHECKS
    - ●● PEOPLE IN LOOP
    - ●● PHYSICAL/ADMINISTRATIVE CONTROL OF SWITCHES
    - ●● EXTENSIVE MESSAGE JOURNALLING
    - ●● LINK ENCRYPTION
    - ●● RESTRICTED USER/NETWORK INTERFACE, I. E., NO USER PROGRAMMING IN SWITCH

WWMCCS COMPUTER INTERCONNECTION VIA AUTODIN I

- PHYSICAL/ADMINISTRATIVE CONTROL GEOGRAPHICALLY DISPERSED, INCOMPLETE
    - ●● WWMCCS SITES GEOGRAPHICALLY DISPERSED
    - ●● NON-WWMCCS SUBSCRIBERS BEYOND WWMCCS CONTROL

- NO DATA SHARING IN COMPUTER SENSE VIA AUTODIN I
    - ●● RECORD TRAFFIC PASSED AMONG WWMCCS SITES, OTHER SUBSCRIBERS
    - ●● NO INTERACTIVE USER PROGRAMMING VIA AUTODIN I

C-14

WWMCCS INTERCOMPUTER
NETWORK (WIN)

- INITIAL OPERATION AS PWIN 1974

- PACKET SWITCHING NETWORK BASED ON ARPANET
  TECHNOLOGY
  - PACKET SWITCHING BY IMPS
  - 50 KB/S TRUNKS

- NETWORK DEDICATED TO WWMCCS SUBSCRIBERS
  - CURRENTLY NINE HOST SUBSCRIBERS
  - ADDITIONAL 11 HOSTS PLANNED BY FEBRUARY 1980
  - NO DIAL-UP ACCESS

- PROVIDES COMPUTER-TO-COMPUTER INFORMATION
  EXCHANGE
  - INTERACTIVE OPERATION
  - NCP, TELNET, TELECONFERENCING, FTP PROTOCOLS

1980 - WWMCCS INTERCOMPUTER
NETWORK (WIN)

C-15
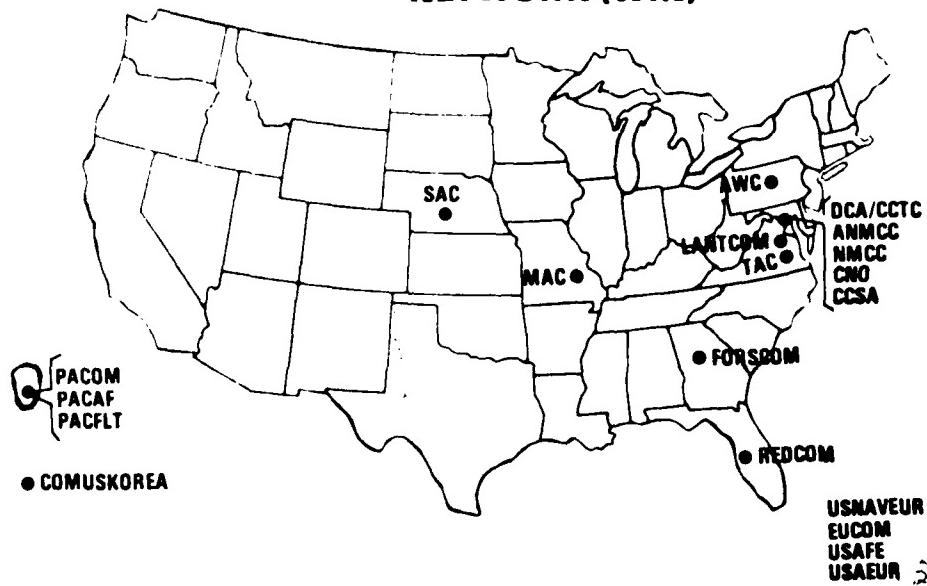
WIN SECURITY

NOT MULTILEVEL SECURE

     SYSTEM HIGH (TS) FOR ALL HOSTS

     PHYSICAL/ADMINISTRATIVE CONTROL OF ALL SWITCHES,

        ACCESS AREAS

        LINK ENCRYPTION

WWMCCS INTERCONNECTION VIA WIN

- PHYSICAL/ADMINISTRATIVE CONTROL GEOGRAPHICALLY DISPERSED, COMPLETE
  - •• WWMCCS SITES GEOGRAPHICALLY DISPERSED
  - •• SUBNET DEDICATED TO WWMCCS COMMUNITY
  - •• ACCESS AREAS CONTROLLED (NO DIAL-UP ACCESS)

- ALL INTERCONNECTED SITES OPERATE SYSTEM HIGH AT TS

- DATA SHARING AMONG INTERCONNECTED SITES

AUTODIN II

- INITIAL OPERATION 1980
- PACKET SWITCHING NETWORK
  - 4 PSN's INITIALLY; EXPANDABLE TO 8
  - PSN's ARE MULTIPLE PDP 11/70's
  - 56 KB/S TRUNKS
- COMMON USER DOD NETWORK
  - MILITARY AGENCIES
  - INTELLIGENCE AGENCIES
  - DOD RELATED ACTIVITIES
  - DIAL-UP ACCESS PROVIDED
- PROVIDES COMPUTER-TO-COMPUTER INFORMATION EXCHANGE
  - INTERACTIVE SERVICE
  - TCP, THP PROTOCOLS; LAYERED FOR EVOLUTIONARY GROWTH

AUTODIN II SECURITY

MULTILEVEL SECURE

- KERNEL TECHNOLOGY IN PSN'S

  FORMAL VERIFICATION OF TOP LEVEL
  SPECIFICATION

  FACILITATED BY RESTRICTED USER/NETWORK
  INTERFACE, I.E., NO USER PROGRAMMING IN PSN'S

- PHYSICAL/ADMINSTRATIVE CONTROL OF ALL PSN'S

- LINK ENCRYPTION

C-17

WWMCCS INTERCONNECTION
VIA AUTODIN II

- PHYSICAL/ADMINISTRATIVE CONTROL GEOGRAPHICALLY DISPERSED, INCOMPLETE
  - •• WWMCCS SITES GEOGRAPHICALLY DISPERSED

  - •• NON-WWMCCS SUBSCRIBERS BEYOND WWMCCS CONTROL
  - •• DIAL-UP ACCESS PROVIDED

- AUTODIN II PROVIDES MLS DATA EXCHANGE - SITES NOT MLS

- DATA SHARING AMONG WWMCCS SITES, OTHER SUBSCRIBERS

LESSONS LEARNED

NO.1 OUR APPLICATIONS "EYES" ARE BIGGER THAN OUR SECURITY TECHNOLOGY "STOMACHS"

- ENORMOUS ADVANCES IN COMPUTER HARDWARE HAVE OCCURRED

- MODEST BUT SIGNIFICANT ADVANCES IN COMPUTER SOFTWARE HAVE OCCURRED

- THESE ADVANCES SPAWNED REQUIREMENTS FOR SOPHISTICATED APPLICATIONS

- THESE APPLICATIONS HAVE LED TO GREATER SECURITY BURDENS

- SECURITY TECHNOLOGY HAS LAGGED

C-18

TECHNOLOGY ADVANCES

HARDWARE

SOFTWARE

SECURITY

TIME
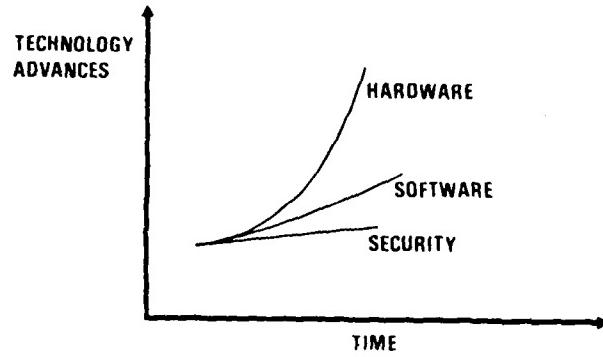
LESSONS LEARNED

NO. 2. SOFTWARE, NOT HARDWARE, DOMINATES LIFE CYCLE COST

- SOFTWARE DEVELOPMENT COST
  (QUICK DESIGN, QUICK CODING, TEST, FIX, TEST, FIX, . . .)

- SOFTWARE MAINTENANCE COST
  (SOFTWARE CHANGE, TEST, FIX, TEST, FIX, . . .)

- SOFTWARE DOCUMENTATION COST
  (INCOMPLETE, INACCURATE, OUT-OF-DATE SYSTEM DESCRIPTIONS)

- SOFTWARE CONVERSION COST
  (DATA AND PROCEDURE)

LESSONS LEARNED

NO. 3. SECURE SOFTWARE MAY BE MORE EXPENSIVE STILL

- INCREASED DESIGN COST

- ADDITIONAL COST OF VERIFICATION AND CERTIFICATION

- INCREASED MAINTENANCE COST FOR RE-VERIFICATION, RE-CERTIFICATION

- PERFORMANCE PENALTY OF UNKNOWN MAGNITUDE

LESSONS LEARNED

NO. 4. SOFTWARE INVESTMENT MUST BE PRESERVED OVER CHANGES IN HARDWARE, REQUIREMENTS

- A. MINIMIZE HARDWARE DEPENDENCIES

  - SPECIFY SYSTEM BY FUNCTION, NOT HARDWARE CHARACTERISTICS

  - MINIMIZE ASSUMPTIONS MADE IN SOFTWARE ABOUT HARDWARE CHARACTERISTICS

  - EVALUATE IMPACT BEFORE OPTIMIZING SOFTWARE FOR PARTICULAR HARDWARE CONFIGURATION

  - USE UBIQUITOUS HIGHER ORDER LANGUAGE(S)

LESSONS LEARNED

NO. 4. SOFTWARE INVESTMENT MUST BE PRESERVED OVER CHANGES IN HARDWARE, REQUIREMENTS

- B. SIMPLIFY SOFTWARE

- • PROGRAM FOR CLARITY, EASY MAINTENANCE, EASY TRANSPORT-ABILITY (NOT DEVELOPMENT SPEED, EXECUTION SPEED, ETC.)

- • SUBSTITUTE HARDWARE FOR SOFTWARE COMPLEXITY WHERE FEASIBLE (BUY MORE MEMORY, FASTER PROCESSOR, ETC.)

- • AUGMENT WITH OTHER APPROACHES WHICH LIMIT THE AMOUNT OF SECURITY RELATED SOFTWARE (e. g., ENCRYPTION)

LESSONS LEARNED

NO. 4. SOFTWARE INVESTMENT MUST BE PRESERVED OVER CHANGES IN HARDWARE, REQUIREMENTS

o C. RATIONALIZE, CONTROL SOFTWARE DEVELOPMENT PROCESS

- • TOP-DOWN DESIGN, IMPLEMENTATION

- • USE SOFTWARE DEVELOPMENT TOOLS

- • EMPHASIZE CLEAR, COMPLETE, CURRENT DOCUMENTATION

LESSONS LEARNED

NO. 5. CAPITALIZE ON COMMERCIAL HARDWARE DEVELOPMENTS

- USE OFF-THE-SHELF HARDWARE WHENEVER FEASIBLE
  (NOT CUSTOM DESIGNED, SPECIAL PURPOSE GEAR)
- EMPHASIZE (AND PAY FOR) MORE TRANSPORTABLE SOFTWARE

CONCLUSIONS

- DEMAND IS INCREASING FOR MORE ADP SUPPORT FOR THE
  WWMCCS VIA NETWORKING

- MULTILEVEL SECURITY CANNOT BE RETROFITTED INTO THE
  CURRENT GENERATION OF WWMCCS ADP

- PROSPECTS FOR THE NEXT GENERATION OF WWMCCS ADP
  ARE EXCITING, BUT SECURITY IS APT TO BE THE MAJOR
  TECHNOLOGICAL LIMITATION

SPECIFIC WWMCCS TECHNICAL OBJECTIVES

- MORE RELIABLE AUTHENTICATION METHODS

- MULTILEVEL SECURITY (FOR HOSTS, AMHS, . . .)

- DISCRETIONARY (NEED TO KNOW) SECURITY AT EFFECTIVE
  LEVEL OF GRANULARITY

- PROTECTION AGAINST DENIAL OF SERVICE

- END-TO-END NETWORK SECURITY

- ALL THE ABOVE AT MODEST COST

C-23

Impact of Computer Security in the Federal Government

J. H. Burrows
Director
Institute for Computer Sciences and Technology
National Bureau of Standards

Introduction

As a co-sponsor and host of this Seminar on Computer Security, I would
like to welcome you to the National Bureau of Standards. As Steve said,
before coming to the Bureau as Director of its Institute for Computer Sciences
and Technology in May of last year, I spent seven years in the Department of
Defense working in many areas in which computers played an important, if not
crucial role. While there, I became more interested in the computer security
problem in general and the Computer Security Initiative in Trusted Computer
Systems in particular. I am very pleased that I am able to participate in
this seminar, both because of my personal experience and interest, but also
because of the official role that we at NBS are playing in computer security.

I want to spend the next few minutes telling you about our perception
of the need for computer security within the Federal Government but not
necessarily within the Department of Defense. We have just heard two
excellent presentations on the need for such security within the intelligence
and communications arms of DoD. In the next two and one-half days, you will
hear a great deal on the technology of computer security and also specific
requirements for this security. Rather than dwelling on the technology or
repeating the general platitudes of computer security, I want to walk through
a case example which we at NBS find very intriguing, not only because of its
obvious need for the technology to be discussed at this seminar, but also
because of the multitude of reasons why this technology may have difficulty
being accepted. Before I begin that walk, I would like to outline our
perspective of a trusted system.

According to the brochure announcing this seminar, a "trusted" ADP
system is one which employs sufficient hardware and software integrity measures
to allow its use for simultaneously processing multiple levels of classified
and/or sensitive information. This personification of a machine requires some
evaluation. A "trustee" in a prison is "trusted" not to break out of prison.
I hope this is not the proper analogy to a trusted computer. A "trustee" in
a real estate loan transaction is a "trusted" third party who knows and
explicitly carries out the policies and procedures that were specified in the
transaction agreement between the two primary parties. The trustee is
required by law to automatically carry out these procedures when triggered by
a specific event. I believe this is the analogy we seek when personifying
a "trusted" computer. The suppliers of data and the users of the data are
the two primary parties in our example. The computer is the third party
"trustee." It must, however, be programmed with the policies and procedures
of protection and control desired by the primary parties. My view is that the
Department of Defense has a "leg up" on the rest of the Federal Government
because of the written existence of these policies and procedures regarding
classified information. The technology of the DoD trusted system will be
applicable but the first requirement for computer security outside DoD is for
the development of a new set of policies and procedures to be implemented in

the trusted system. This is no trivial task. In the extremely simple area of classified information (not involving money or property) and a single unified goal, DoD struggled for over five years to come up with their written policy and have yet to close on the parameters of a certification policy acceptable to all of DoD.

Case Example of Computer Security Requirements

I am sure you all are aware of the existence of the Environmental Protection Agency. I am not sure if you know of their very broad role as dictated by Congress and EPA's establishing Presidential order. I have chosen to use EPA as an example of the need for computer security in the Federal Government and to outline the multitude of conflicting data processing security requirements. EPA has informally requested the assistance of NBS in specifying reasonable security provisions which can best satisfy these requirements. For this reason, I am able to outline some of the problems as seen by EPA in this area.

EPA was established by Presidential Order in 1970. At that time, 15 environmental control programs were unified in a single regulatory agency. EPA is charged with mounting an integrated, coordinated attack on the environmental problems of air and water pollution, solid waste management, pesticides, radiation, noise and toxic substances. Since its establishment, Congress has added to this list of responsibilities. EPA is specifically required to set standards for the limits of pollution, enforce compliance with these limits and monitor the progress of reducing pollution even if below the established limits.

Organizationally, EPA consists of six primary offices reporting to an Agency Administrator. Best known are the Office of Water and Hazardous Materials, Office of Air and Waste Management, and the Office of Toxic Substances. Thirteen different major Congressional Acts govern the actions of just these three offices. Some examples of their regulatory responsibilities will serve to demonstrate their need for complete and accurate data and trusted data processing capabilities.

o   EPA is required to conduct research on all aspects of water pollution and control dumping of all pollutants (including radioactive waste) into all navigable waters. Any U.S. citizen has the right to take legal action against a water polluter.

o   EPA is required to set safe drinking water regulations and enforce those regulations if each state does not or cannot. Again, any citizen has the expressed right to bring civil suit against any public water system or any Federal agency (including EPA) in violation.

o   EPA is required to set air quality standards for all air pollutants and assisting states in meeting those standards. Citizens are specifically authorized to take necessary legal actions against private or Government officials failing to meet provisions of the Clean Act.

o   EPA is required to develop programs for testing, monitoring and regulating chemical substances and pesticides. Manufacturers must register with EPA all insecticides, herbicides and fungicides intended for sale in the U.S. and notify EPA at least 90 days before the manufacture of any chemical intended for commercial purposes.

o EPA is required to collect, maintain, protect and process accurate and
   complete data in support of these activities.

EPA estimates that 35 billion characters of data will be maintained in the
data bases of its two computer centers - one in North Carolina and one in Washington -
in 1980 with a fivefold increase expected in 1990. A completely new computer
facility is being designed to support all the activities of EPA. Presently
intended for first operation in the mid 80's, the system will be designed according
to the goals of OMB Circular A-109 to be useful and cost-effective for a life cycle
of 10-15 years.

Some interesting security, integrity and privacy issues which EPA must
address include:

(1) Manufacturers must submit extensive reports to EPA on a regular basis.
EPA plans to allow the manufacturers to supply the data in automated form (tape
or disk) in order to save the cost of reentering the data into its computer when
the data is already in a computer. One consideration is to allow direct access
to EPA's data bases by the manufacturer in order to improve the timeliness of
data. If this is allowed, how can access be restricted to only that data
previously submitted by that manufacturer?

(2) Each of the 50 states has many rights and responsibilities under the
various acts coordinated by EPA. Great competition exists among states and
their legislators for Federal dollars. What data should states be allowed to
access? As many funding initiatives are based on states' programs, what controls
can be placed on this data from Congressional and public interest inquiries?

(3) Members of the scientific community have had great interest in using
data maintained by EPA. In one instance in North Carolina, technical consultants
and world-famous employees refused to use picture badges issued for automated
door locks which controlled access to the research laboratories. How will
scientists used to having access to scientific informaton react to access controls
uniformly enforced by the "trusted" ADP system?

(4) An EPA spokesman estimated that there are now 40 lawsuits by manu-
facturers filed against EPA seeking less stringent environmental controls and
35 lawsuits by public interest groups filed against EPA seeking more stringent
environmental controls. Attorneys from both sides commonly use the data bases
maintained by EPA to support their side of such cases. Attorneys for both sides
now resort to attacking the integrity of EPA's data bases and data processing
(including timeliness, accuracy, and completeness of the data) when their case
is in jeopardy. How can the integrity of data and data processing be sub-
stantiated in a court case?

EPA's data processing and security requirements were particularly
interesting to me because they demonstrated that a trusted and provable secure
computer system was needed outside both DoD and money handling agencies such as
IRS, Treasury, Social Security and the Federal Reserve System. Unauthorized
disclosure of classified information and financial fraud are both understood
problems which have legal histories and penalties. The Privacy Act of 1974
provides penalties for the intentional disclosure of personal information. The

draft "Federal Computer Systems Protection Act of 1979," introduced by Senator Ribicoff as S.240, makes it a crime to use, for fraudulent or other illegal purposes, any computer owned or operated by the United States. The bill provides a maximum penalty cf 15 years in prison and/or $50,000 fine for willful and unauthorized accesses or attempts to access a computer network for the purpose of altering or obtaining programs and data. Without the passage of this law, however, EPA has no legal basis to enforce access controls imposed on the very sensitive data supplied by manufacturers, states, cities, and private organizations. Yet their most important role is to maintain accurate and up-to-date data and be capable of validating its integrity, securely processing it for various applications, and assuring its suppliers that the data has only been used for its intended purposes. The computer must become the "trustee" in this environment and a "trusted" system is necessary according to the security requirements of EPA.

NBS Computer Security Role

NBS has the responsibility of developing Federal Information Processing Standards to improve the utilization of computers in the Federal Government. Historically, these standards have included the ASCII character standard, the Cobol language standard, magnetic tape standards and computer-peripheral interface standards. Now, the Office of Management and Budget has given NBS responsibility for developing computer security standards and guidelines in Transmittal Memorandum 1 of OMB Circular A-71. I would like to spend a few minutes outlining our computer security program and how we are responding to added impetus to our security program.

ICST first established a computer security program in 1972 which has:

(1) Issued six Federal Information Processing Standards or Guidelines devoted to computer security.

(2) Published numerous technical articles and special publications in computer security technology.

(3) Worked with the American National Standards Institute in developing voluntary ANSI standards in security of communications and financial transactions.

(4) Established a five year program to issue Federal standards in the following computer security areas:

      o Risk Analysis

      o Contingency Planning

      o Security Auditing

      o Personal Identification

      o Network Security

      o Data Encryption

      o Applications Program Development

During the first seminar in this series, Dr. Willis Ware of the Rand Corporation, in his keynote address, challenged ICST to "step out smartly in specifying the performance requirements of a secure operating system plus the administrative, procedural and physical environments in which it is embedded." Willis stated that "within the Civil Government Sector, only ICST has a chance of handling the computer security problem." Willis did not say how small that chance was. In support of ICST, Willis did subsequently request OMB to provide us reasonable resources to meet this challenge. To date, we have received no additional resources in this area. In addition, let me outline what I perceive as a set of necessary conditions for anyone to meet his challenge.

(1) A uniform master structure for policies for the protection of data within all computer systems of the Federal Government must be established.

(2) The technology for trusted systems must be researched and developed.

(3) A willingness to design, build and maintain "trusted" computer systems must be made by private industry when the technology becomes available.

(4) A commitment for the procurement and use of "trusted" computer systems (when they become available) must be made by all Federal ADP organizations when security is needed and physical protection does not suffice.

(5) A standard for the functional requirements of a trusted system must be developed and issued.

(6) A validation service which tests commercial implementations of the trusted system against the standard must be established.

(7) Procurement regulations which make the procurement of trusted systems simple must be issued.

Dr. Ware called for NBS to follow the same road in developing a trusted computer system standard that was followed by NBS in developing a Data Encryption Standard. This we will do. We requested the assistance of industry and NSA in specifying and evaluating the technology for the Data Encryption Standard. I hereby request the similar assistance of industry and DoD to research, implement, test and evaluate the technology of a "trusted" operating system. NBS can then begin the drafting and coordinating of a Federal Information Processing Standard. We can also initiate a cooperative effort with DoD in establishing a validation service and with GSA in modifying the procurement regulations to make it easy for a Federal agency to obtain a trusted system.

The ultimate impact of computer security in the Federal Government cannot be estimated at this time. Even the scope of computer security has not been completely specified. Impact will have both positive and negative

aspects. Processing reliability will definitely be improved because of security provisions. Data integrity will definitely be improved, perhaps to the point where courts can be convinced of its correctness. Trust of the suppliers and users of data will be improved when the computer becomes a true "trustee." However, costs will be incurred. People will become frustrated when they can no longer access the computer system like they could before. Programmers became frustrated (for a short while) when they were barred from the computer room. Money must be spent -- both for initial and operational costs. The acceptability of a trusted computer system will be based on these factors.

NBS is pleased to provide a forum for DoD to discuss its computer security initiative with you. We will also be happy to "step out smartly" in support of computer security technology and promulgating reasonable and effective standards in this area as the technology becomes available.

COMPUTER SECURITY INTEREST IN THE PRIVATE SECTOR
by
Edwin L. Jacks
General Motors Information Systems and Communication Activity

Good morning, Ladies and Gentlemen. When Dr. Steve Walker invited me to speak on the "Computer Security Interest in the Private Sector", it gave me a welcomed chance to relate our viewpoint on computer security in GM to the computer security viewpoint as expressed by the Department of Defense Computer Security Initiative Program.

While the title to my talk is "Computer Security Interest in the Private Sector", much of my speech will be on the computer security activities within GM. Based on the internal interest now being shown by many companies on the subject and discussions I have had with data processing managers of other companies, our interests are, I believe, similar.

Computer systems have become a basic resource used in the operation of a business. The exception today is the non-use of computers in a business function - not the use of computers. Their basic utility has been clearly proven. Their penetration is into every aspect of business. As reported by The Diebold Group, Inc., expenditures for information systems have been approximately 1% of the sales revenues for large companies, as a national average, for the last 10 years. In GM, one would be hard-pressed to find an aspect of the business not using computers. The reasons for using computers are diverse: they may be economic - a means to reduce operating cost; or business effectiveness - a means to better operate the business. The private sector is using computers to aid design, engineering, and manufacturing of its products; to aid the ordering, marketing, and distribution of its products; to aid in the accounting, purchasing, and invoicing processes; and to aid in the personnel and legal activities of its business.

The end effect is an extensive business dependency on computer systems.

That dependency has led the private sector into having a basic concern about the security of computerized business systems. From a strictly technical viewpoint, the only thing that can happen to data is the unauthorized disclosure, modification, destruction, or delay.

The private sector concern, however, is with the consequences of security failure - for example, loss of production, loss of assets, loss of confidentiality, and loss of customer service.

Information System Security

As I understand the DOD initiative, its objective is ". . . the widespread availability of trusted computer systems." Those are systems with sufficient hardware and software integrity measures to allow their use for simultaneously processing multiple levels of classified or sensitive information. The broad business concern is the prevention of failure for any reason of the information system portion of a business system. From a DP management viewpoint in the private sector, the security objective is to provide business managers with trusted information systems. One definition that we have used in GM is that, "Security is knowing your business procedures, being confident of their operational effectiveness, and being sure they are in place."

This definition places a positive emphasis on security. It emphasizes the need for the business manager to understand the computerized portions of his business systems. At the same time, it places a responsibility on the system developer to communicate clearly with the business manager and to accurately implement systems. The operational effectiveness requirement points up the need for business managers to evaluate how well a system meets business objectives. It identifies his responsibility and accountability for a system. And, finally, the "being sure the systems are in place", addresses issues of auditability and integrity.

We believe that if the business manager clearly understands the computer procedures being used, can measure their operational effectiveness, and is sure they are in place, then we will indeed have trusted business systems.

Now, as this audience knows well, from the rigorous mathematical viewpoint, the development of trusted computer systems is at the front edge of computer technology. However, thousands of information systems are working, and working well and are, in a practical business sense, trusted.

If this is the case, then how does the DOD initiative fit into the private sector information systems security interest? To address that point, the presentation will first cover various background aspects of the security activities within GM; second, discuss a formulation of security as maintaining, under adverse conditions, three independent parameters: availability, integrity, and confidentiality; third, discuss design concepts for secure systems - in particular, responsibility assignment, security continuity, separation of incompatible functions, and minimization of failure impact; and, finally, relate the preceding items to the DOD initiative. In particular, we find trusted computer system concepts necessary but not sufficient for the private sector information system security.

E-2

Security - A Pragmatic Approach

GM is a decentralized organization with major data centers in almost every division and in several staff functions. The Data Processing Manager at each of these centers has a specific responsibility for information systems security. In that most of these centers have evolved from the Financial operations in GM, they have a long history of careful control and handling of data. Their security initiatives are both long standing and pragmatic. They install those procedures and practices which economically protect their information systems.

The focus point for a data processing installation's security is the Information System Security Officer - an ISSO. Corporate policy requires each data processing installation to have an ISSO. He - or she, as the case may be - is the one responsible for administering an on-going security program in the installation.

That program would include physical security, operational controls, and disaster recovery plans. The operating system would be expected to have integrity at least equivalent to IBM's MVS system. Any support software required must not violate or be detrimental to the system integrity. A resource access control program with individual identification and password authentication would be in use with the operating system.

The process of program specification and development would be under careful control. Controls would include review and approval of program specification by the application manager. If appropriate, divisional audit people would also review the specification. Particular attention would be paid to the processes of final testing and promotion to production. Tight - and auditable - controls would be in place to ensure that the tested and authorized programs for production are in production. Controls would be in place to ensure that the development programmer could not change the production program source or object code file. In addition, either sample or one hundred percent inspections (depending on the sensitivity of the program) would be in place to verify the correctness of the program relative to the application specifications.

And finally, within each division, GM has internal auditors to provide assurance that information system security procedures are in place.

Security Initiative - A Staff Stimulant

In addition to the divisional data processing security activity in GM, there are vigorous security activities in the GM Coroporate Audit Staff and the GM Information Systems and Communication Activity. The Audit Staff has a data processing

audit group that consists of experienced data processing operations, systems development, and operating systems personnel. These people, with audit thoroughness and data processing expertise, audit the divisional security activities. In addition, they aggressively follow the state-of-the-security-art to ensure that divisions are using economically effective means to improve security.

The Security group within the GM Information System and Communication Activity serves several functions.

One is consulting with divisions on their overall security programs. Another is reviewing the security of various computer hardware and software products. A third is coordinating the security program activities of the divisional Information System Security Officers. A final function is the understanding and development of security concepts to support GM's short and long term needs for data security.

Security Objectives

In working with divisions, we at the staff level get clearly challenged by the problems faced by the divisional Data Processing Managers in designing their security system. For instance, obviously sensitive and critical data should be handled in a secure fashion. However, in GM and most private sector companies, the words 'sensitive' and 'critical' are informally defined as classifications of data. Often those informal definitions pre-date the computer age. The formal classification of data requires identification of formal procedures for handling each class of data, a requirement for individual clearance for a given class of data, and the assignment of the classification to each data element. Little utility is seen in the private sector for this formal approach to classification. In studying the classification of data issue, John Maier of my staff, working with divisional ISSOs, was forced to ask, "What are we trying to accomplish with classification?"

Our prime interest was to establish standard classes for both information systems and data. This was to be done so that we could say that a given application is of a given class and hence should be handled with the rules for that class.

Now what would the rules for a class be? Who has clearance to see the data? In GM we, in general, don't clear people; we trust them and give them responsibilities so that, at best, clearance has to do with need to know.

Would a rule have to do with timeliness of data and, for instance, after a disaster, allowable recovery time?

The study found no practical economic and effective classification scheme. However, a very positive result occurred: it became clear that, for a given system, three independent security objectives should be specified. Those objectives were identified as:

Availability – a system's readiness for use; a system's ability to receive, to process, and to send information in a timely and effective manner; a system's ability to recovery from unwanted events.

Integrity – a system's soundness; a system's accuracy, its correctness, and its completeness.

Confidentiality – a system's closeness; a system's control over the access, visibility, and dissemination of information.

The process of establishing availability, integrity, and confidentiality requirements was found to relate to some general questions.

1. Who uses this information resource?

2. How is it used?

3. Where is it sent?

4. Is this information resource primarily of a financial, manufacturing, engineering, sales, marketing, personnel, or logist cal nature?

5. Are there legal requirements for the processing, storage, and communication of this information?

6. Is there existing GM policy which governs the way this information is processed, stored, and communicated?

The particular concerns of availability can be identified by examining consequences of loss or delay; for example:

1. What would be the consequences of not having this information or processing resource?

2. What would be the consequences of delay of this information?

3. How long can you "get by" without it?

4. How current must it be?

E-5

5. Where must it be available?

6. Are there alternative means of processing or are there alternative sources of the information?

7. How long must it be accessible?

8. How quickly must it be accessible in usable form?

The concerns for integrity can be identified by examining consequences of incorrectness, such as:

1. Does this application account for a business operation? What would the consequences be if this accounting was incorrect?

2. Does this application direct a business operation? What would be the consequences if this operation was directed incorrectly?

3. Does this application specify or control the manner in which a process is performed? What would be the consequences if this specification was incorrect?

4. Does this application predict future environments or events? What would the consequences be of incorrect prediction?

5. Does this information resource specify the design or engineering of products? What would the consequences be of incorrect product specification?

6. Does this application contain business transactions which should not be performed by a single individual?

7. Could manipulation of this resource result in personal gain?

The concerns of confidentiality can be identified by examining consequences of improper disclosure. For instance:

1. Could improper disclosure of this information resource result in a breach of personal privacy?

2. Could this information be considered "inside material"?

3. Could an individual personally profit from knowledge of this information?

4. Could an individual personally profit from improper disclosure of this information?

E-6

5. Who has a legitimate business "need-to-know" this information?

6. Could GM's competition benefit from knowledge of this information?

7. Which information in the system could produce undesirable consequences if disclosed to:

    a) other departments?

    b) other divisions?

    c) the Corporation at large?

    d) the public at large?

From answers to these questions comes specific information on the requirements for availability, integrity, and confidentiality, as well as the consequences of not maintaining the stipulated requirements. This information provides a security objective for a given application.

You will note that most data classification schemes tend only to address the need for various levels of confidentiality. The work on secure computer systems is essentially an effort to provide confidentiality by having proven design and implementation integrity. From an application standpoint, the availability and integrity objectives are only partially improved by the secure computer system work.

Rather than formally classifying data, then, our objective is to provide for each application an appropriate level of availability, integrity, and confidentiality.

Security Design Concepts

In addition to being concerned about clear security objectives, we have been concerned about the design process for secure systems. In that our security objectives do not relate to a formal data classification structure or security model, we must in some manner see the security objectives do get incorporated into systems. Furthermore, the security objectives for a system need to be combined with the system's functional objectives in a manner which achieves the security objectives with minimum change to the system functional objectives. It would also be desirable for the design process for secure systems to fit in with current good system design practices.

Systems can be organized in many different ways to meet functional objectives. For example, systems may be organized so that the transactions being handled by the systems are handled in

minimum   time;   they   may   be   organized   for   minimum
inter-connections between system components;   or,   they   may   be
organized   to obtain high performance by the people that are part
of the system.

The design process usually arrives at a system   organization   and
function   which   is   a   compromise   between objectives.   Security
objectives   are   often   seen   as   being   in   conflict   with   the
functional   objectives   for   a   system.   The conflict may be best
handled   early   in   the   design   process   by   deciding   on   the
availability,   integrity   and confidentiality objectives and then
integrating   those   objectives   into   the   overall   design
consideration for a system, and into the design of each component
of   a   system.   The   components   may   well   be complex disparate
elements: people,   programs,   and   hardware.   Even   with   the
extensive   differences   between   elements,   are   there   design
principles, particularly relevant to security, which may be   used
to guide the design process?   Four principles to consider are:

First,   Responsibility Assignment: the security function must be
specified for each and every component of a system.

Second, Security Continuity: the interface between components of
a system must be identified   and   must   be   consistent   with   the
security function of each component.

Third,   Separation   of   Incompatible Functions: system functions
should be placed in separate components of the   system   if   those
functions,   when   performed   separately,   may serve as a check or
control on each other and, when   performed   together,   can   cause
system security failure if the component fails.

Fourth,   Minimization   of   Failure   Impact: given   alternative
satisfactory system designs, select the system organization which
minimizes the detrimental impact on the system when   a   component
fails.

The   above   design   concepts   for   security   inter-relate   the
identification of security   functions   for   each   component,   the
interfaces   between components, and the overall organization of a
system.

The first two design principles, responsibility   assignment   and
security   continuity,   are   important   because   they   address the
unique characteristic of security as a distributed property of   a
system,   and   yet a property in which any component or collection
of components of a system may cause the total system to not   meet
explicit   stated   security   objectives.   The   third   principle,
separation of incompatible functions, recognizes   that   component
failure   may   occur,   and that, if possible, the system functions
should be designed to prevent individual component failures   from
causing   system failures.   The probability of any given component

E-8

failing, as well as the consequences of failure, may be used to determine to what extent functions need to be separated. The fourth principle, minimization of failure impact, is a combination of Murphy's Law, "When things can go wrong they will go wrong," and a decision theory viewpoint of, "When selecting among alternative satisfactory courses of action, select the one which will minimize your maximum regret when things go wrong."

Do Today's Systems Measure Up?

Data processing security has pragmatically evolved during the last 10 to 15 years. It did not come from carefully thought out security objectives, nor from carefully applied system security design principles. However, the security measures in use today can be measured against those objectives and principles. In doing that, it becomes clear that there are weaknesses. Commercially available computer systems today only in part support the building of secure information systems. The security objective of maintaining availability, integrity, and confidentiality under adverse conditions are not inherent in most commercial systems; and the design principles of responsibility assignment, security continuity, separation of function, and minimization of failure impact were, in general, not used in building most commercial systems.

For example, how can you mix large and small computers in a system and have functional responsibility assignments if there is no integrity in the computer operating systems? How can you achieve continuity of responsibility in a computer system when the operating systems, data communication systems, data base system, and program library system have independent approaches to access control? The problems are well-known to this audience.

I would like to close by noting that the definition of security I used earlier in the talk is in many ways parallel to aspects of the Computer Security initiative program. The first part was "knowing your business procedures."

As you are well aware, the process of constructing trusted computer systems starts by focusing on a schema for construction of computer systems which have proveable properties. In effect, that permits a system design team to say, "What we believe our system does is exactly what it does." Under those conditions you will truly 'know your system'.

The second part of the definition was "being confident of their (the procedures) operational effectiveness." The performance and efficiency of the schemas for producing proveable systems is still unknown. But, more importantly, will the procedures being developed satisfy the security objectives of availability, integrity, and confidentiality? With the focus on confidentiality, has availability been compromised? At this time

I believe the answer is unknown.

Finally, the last part of the security definition was ". . . being sure they (the procedures) are in place."

In this regard, the proveability that a desired system property is in a system will be a big step forward; and, of course, having a trusted computer system as the base for an application system is necessary for trusted information systems.

As I indicated at the beginning, businesses depend extensively on computer systems. That dependency will be viewed as almost insignificant when compared to the business dependency ten years from now. New, large systems will be evolving that will be pervasive throughout business organizations. For those systems to be secure business systems, the concepts of trusted computer systems being developed in the Department of Defense Computer Security Initiative Program are clearly needed.

Acknowledgement

Many of the ideas contained in this paper have evolved from the excellent working relationship between the GM Information Systems and Communication Activity Staff (GMISCA) and GM Divisional Information System Security Officers(ISSOs). In particular, the contributions of John Wiest of GM Assembly Division, and of John Maier and Ashby Woolf of GMISCA are acknowledged.

01/24/80

STATUS OF THE DOD

COMPUTER SECURITY INITIATIVE

STEPHEN T. WALKER
CHAIRMAN
DOD COMPUTER SECURITY
TECHNICAL CONSORTIUM

DEPARTMENT OF DEFENSE

COMPUTER SECURITY

INITIATIVE

... TO ACHIEVE WIDE SPREAD
AVAILABILITY OF TRUSTED
COMPUTER SYSTEMS

COMPUTER SECURITY EVOLUTION

1968 – 1973   "TIGER TEAM" PENETRATIONS

1972   AF/ESD STUDY – REFERENCE MONITOR CONCEPT

1973 – 1975   ESD/MITRE DEVELOPED PROTOTYPE SECURITY KERNEL ON 11/45, APPLIED CONCEPT TO MULTICS DESIGN

1974 – 1975   UCLA DEVELOPED SECURITY KERNEL FOR A VIRTUAL MACHINE MONITOR ON 11/45

1975   ARPA COMPUTER SECURITY WORKING GROUP FORMED

F-3

COMPUTER SECURITY EVOLUTION

1971 - 1975 BELL LABS UNIX OPERATING SYSTEM DEVELOPED - WIDELY USED

1976 UCLA AND MITRE BEGAN IMPLEMENTING SECURE UNIX PROTOTYPES

1976 SDC BEGAN DEVELOPMENT OF KERNELIZED VERSION OF IBM VM370 OPERATING SYSTEM (KVM)

1977 PRODUCTION VERSION OF SECURE UNIX OPERATING SYSTEM INITIATED (KSOS)

1978 DOD COMPUTER SECURITY TECHNICAL CONSORTIUM

1978 DOD COMPUTER SECURITY INITIATIVE

F-4

COMPUTER SECURITY EVOLUTION

1973   ESD/MITRE
       SECURITY MODEL
       SECURITY KERNEL

1974   MULTICS

1975   UCLA
       SECURITY
       KERNEL

1976   SECURE   SECURE
       UNIX   UNIX
       PROTOTYPE   PROTOTYPE

1977   FACC
       KERNELIZED
       SECURE
       OPERATING
       SYSTEM
       (KSOS-11)

HONEYWELL
SCOMP

KSOS-6

SDC
KERNELIZED
VM370

SRI
PSOS STUDY

1980s TRUSTED SYSTEM APPLICATIONS
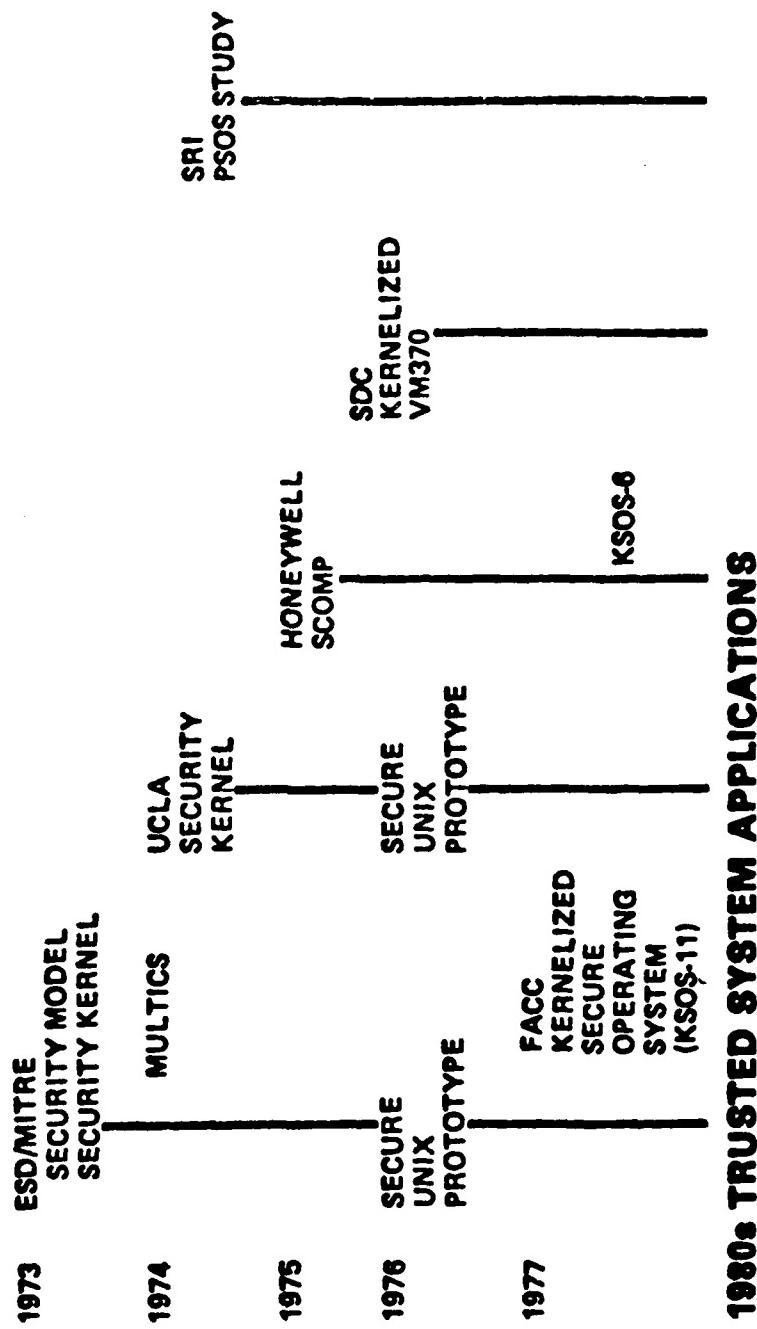
KERNELIZED SECURE OPERATING SYSTEM (KSOS)

"SECURE UNIX" *

1976 – 1977	– UCLA AND MITRE SECURE UNIX PROTOTYPES
AUG 1977	– COMPETITIVE PROCUREMENT, TWO DESIGN PHASE CONTRACTS TRW, FORD AEROSPACE & COMMUNICATIONS CORP.
MAY 1978	– IMPLEMENTATION PHASE CONTRACT: FORD AEROSPACE
MAY 1980	– ALPA TEST SITES
AUG 1980	– BETA TEST SITES
LATE 1980	– AVAILABLE AS SUPPORTED PRODUCT

*BELL SYSTEM TRADE/SERVICE MARK

KERNELIZED VM370

- CERTIFIABLY SECURE VERSION OF IBM VM370 OPERATING SYSTEM

- GUARANTEE SEPARATION OF VIRTUAL MACHINES PROVIDED BY VM370

- 3 YEAR EFFORT, BEGUN IN 1976

- INITIAL DEMONSTRATION — OCTOBER 1979

- ALPHA TEST SITES — SPRING 1980

- AVAILABLE AS SUPPORTED PRODUCT — FALL 1980

- WORK PERFORMED BY SYSTEM DEVELOPMENT CORPORATION

APPROVAL FOR DOD USE

TODAY

DEVELOPMENT GROUP

POLICY

DODD 5200.28
....

PHYSICAL
ADMINISTRATIVE
PERSONNEL

COMPUTER SECURITY

TEMPEST
COMSEC

SPECIFIC SYSTEM REQUEST

DESIGNATED APPROVING AUTHORITY

INDIVIDUAL INSTALLATION APPROVAL

R
E
Q
U
I
R
E
M
E
N
T
S

FOR USE OF ADP PROCESSING CLASSIFIED INFORMATION

APPROVAL FOR DOD USE

INDUSTRY

"LABORATORY EVALUATION"

"EVALUATED PRODUCTS LIST"

POLICY

(DODD 5200.28 ....)

SOON ~1982

DEVELOPMENT GROUP

SPECIFIC SYSTEM REQUEST

DESIGNATED APPROVING AUTHORITY

INDIVIDUAL INSTALLATION APPROVAL

REQUIREMENTS

FOR USE OF ADP PROCESSING CLASSIFIED INFORMATION

EVALUATED PRODUCTS LIST

CLASS	TECHNICAL FEATURES	EXAMPLES	POSSIBLE ENVIRONMENTS
1	—	MOST COMMERCIAL SYSTEMS	DEDICATED MODE
2	FUNCTIONAL SPECIFICATION REASONABLE PENETRATION RESULTS	"MATURE" "ENHANCED" OPERATING SYSTEM	BENIGN, NEED TO KNOW ENVIRONMENTS
3	REASONABLE MODERN PROGRAMMING TECHNIQUES LIMITED SYSTEM INTEGRITY MEASURES	MULTICS	AF DATA SERVICE CENTER TS-S
4	FORMAL DESIGN SPECIFICATIONS SYSTEM INTEGRITY MEASURES		NO USER PROGRAMMING TS-S-C
5	PROVEN DESIGN SPECIFICATIONS VERIFIABLE IMPLEMENTATION LIMITED COVERT PATH PROVISIONS	KSOS KVM	LIMITED USER PROGRAMMING TS-S-C
6	VERIFIED IMPLEMENTATION AUTOMATED TEST GENERATION EXTENDED COVERT PATH PROVISIONS REASONABLE DENIAL OF SERVICE PROVISIONS		FULL USER PROGRAMMING TS-S-C

3459

F-10

COMPUTER SECURITY INITIATIVE

I. **EDUCATION PHASE**
PUBLIC SEMINARS/WORKSHOPS

II. **SPECIFICATION PHASE**
DRAFT | DOD COORD. | INDUSTRY COORD. | REVIEW AND ENHANCEMENT

III. **EVALUATION PHASE**
INFORMAL

KSOS – 11
KVM
KSOS – 6
UNIVAC

FORMAL

INDUSTRY
SUBMITTED
SYSTEMS

"EVALUATED PRODUCTS LIST"

1978   1980   1982   1984

F-11
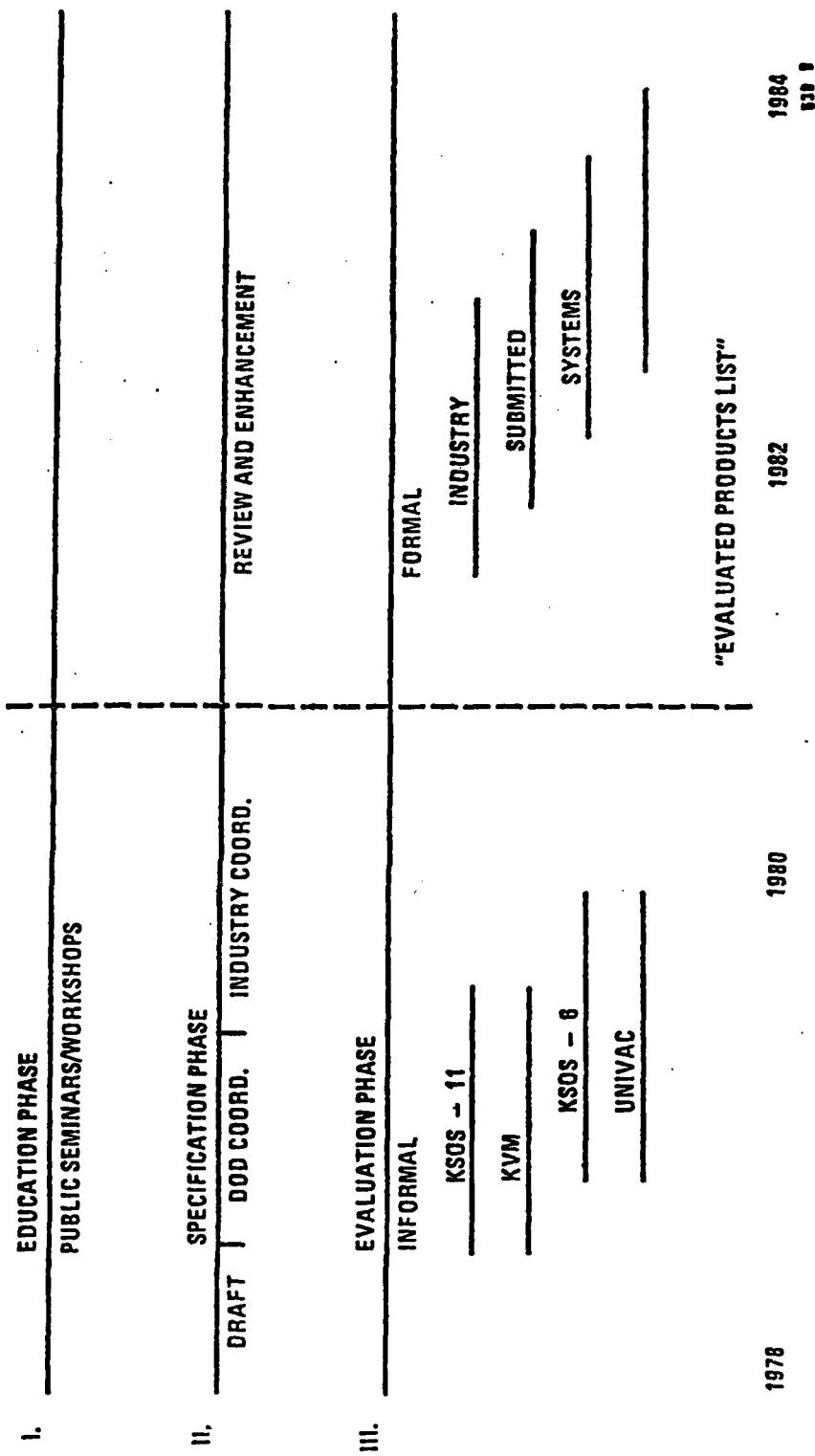
COMPUTER SECURITY (COMPSEC)
IMPACTS ON NEAR TERM SYSTEMS

BY
CLARK WEISSMAN
SYSTEM DEVELOPMENT CORPORATION

PRESENTED AT
SECOND SEMINAR ON DOD COMPUTER SECURITY
INITIATIVE PROGRAM
15-17 JANUARY 1980, NBS GAITHERSBURG, MARYLAND

System Development Corporation

A DECADE OF COMPSEC TECHNOLOGY
FUELS GROWTH IN 1980'S

TECHNOLOGICAL PROGRESS IN:

1. COMPSEC POLICY

   - INFORMAL DOCTRINE & REGULATIONS

   - FORMAL MATH MODELS

2. PROTECTION MECHANISMS

   - COMPSEC REQUIREMENTS

   - ARCHITECTURALLY TRUSTED APPROACHES

3. COMPSEC PRODUCT ASSURANCE

   - CONFORMITY OF PRODUCT → SPEC → POLICY

   - CREDIBILITY OF EVIDENCE

System Development Corporation

COMPSEC POLICY IS THE FOUNDATION OF THE NEW TECHNOLOGY

1. INFORMAL DOCTRINE, REGULATIONS, STANDARDS

    *   DOD 5200.28 SECURITY REQUIREMENTS FOR ADPS

    *   AFR 300.8 ADPS SECURITY POLICY, PROCEDURES, AND RESPONSIBILITIES

        *   COMPSEC PROGRAM OFFICE

        *   DESIGNATED APPROVING AUTHORITY (DAA)

        *   SYSTEM SECURITY OFFICER (SSO)

        *   CLASSIFIED MODES OF OPERATION

    *   AFR 300.13 PERSONAL DATA (PRIVACY) IN ADPS

    *   DCID 1-16 COMPARTMENTED INTELLIGENCE DATA

    *   OMB A71 THREAT & RISK ASSESSMENT IN ADPS

    *   NBS DES UNCLASSIFIED DATA ENCRYPTION STANDARD

System Development Corporation

COMPSEC POLICY IS THE FOUNDATION OF THE NEW TECHNOLOGY (CONT'D)

2. FORMAL COMPSEC POLICY MODELS

    *   SECURITY CONDITION - WEISSMAN - '69 FJCC

    *   ACCESS MATRIX - GRAHAM & P. DENNING - '72 SJCC

    *   T.H. CONFINEMENT - LAMPSON - '73 CACM

    *   *- PROPERTY - BELL & LAPADULA - '73/'74 MITRE

    *   INFO FLOW - D. DENNING - '76 CACM

    *   SPECIAL, INA JO FORMAL SPECIFICATION - MITRE, SRI, SDC, FACC - PRESENT

    *   APPLICATION-SPECIFIC POLICIES - FUTURE - 1985

System Development Corporation

ARCHITECTURALLY TRUSTED APPROACHES
FOR SECURITY ENFORCEMENT MECHANISMS
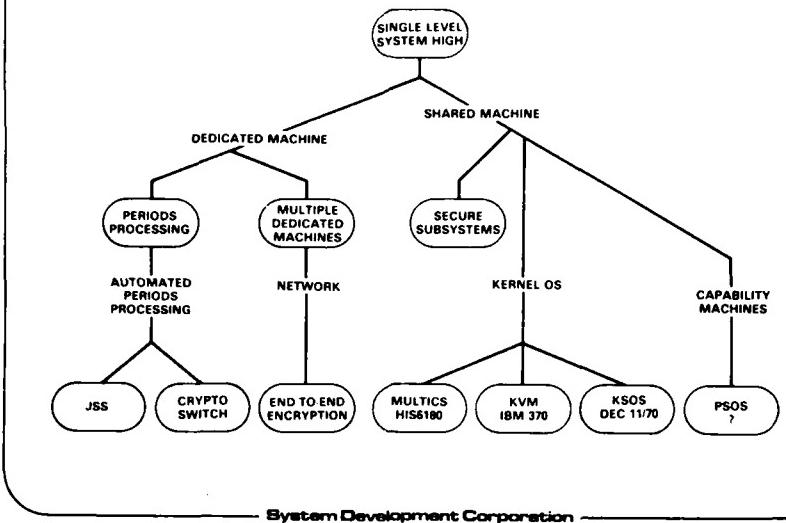
System Development Corporation

CHARACTERISTICS OF
ENFORCEMENT MECHANISMS

1. PERIODS PROCESSING (P P.)

   • BASICALLY SINGLE APPLICATION ("COLOR")
     PER PERIOD

   • LABOR INTENSIVE, SLOW COLOR CHANGE

   • BREAKS OPERATIONS CONTINUITY

   • USUALLY UNSHARED UNDERUTILIZED CPU

   • NO RISK, NO RUNTIME OVERHEAD

   • CURRENT PRACTICE

2. JOB STREAM SEPARATOR (JSS) AND CRYPTO SWITCH

   • AUTOMATIC, FAST COLOR CHANGE — TECHNOLOGY
     INTENSIVE

   • TRUSTED PROCESSOR REQUIRED

   • FUTURE DEVELOPMENT IN PROCESS

System Development Corporation

G-3

CHARACTERISTICS OF
ENFORCEMENT MECHANISMS (CONT'D)

3. SECURE NETWORKS-END-TO-END ENCRYPTION (E^3)
   - MULTI-LEVEL NETWORK
   - TRUSTED DEVICES AND PROCESSORS NEED TO BE DEVELOPED
   - NBS-DES AVAILABLE
   - TRUSTED E^3 PROTOCOLS NEED TO BE DEVELOPED
   - E^3 IS A TRUSTWORTHY TECHNOLOGY
   - COST EFFECTIVE TECHNOLOGY
   - USEFUL FOR AUTHENTICATION AND ACCESS CONTROL
   - FUTURE DEVELOPMENT IN PROCESS

4. SECURE SUBSYSTEMS (S^3)
   - LIMITED USE, TRANSACTION DMS (TDMS)
   - TRUSTED, MULTI-LEVEL TDMS
   - UNTRUSTED OS
   - ONLY TDMS USERS ON OS
   - FUTURE DEVELOPMENT IN PROCESS

System Development Corporation

CHARACTERISTICS OF
ENFORCEMENT MECHANISMS (CONT'D)

5. SECURITY-KERNEL BASED OS
   - FLAW-BY-FLAW REPAIRED OS UNTRUSTWORTHY
   - TRUSTED, MULTI-LEVEL OS WITH KERNEL
   - KERNEL ENFORCES SECURITY POLICY
   - TAMPER PROOF KERNEL
   - KERNEL ALWAYS INVOKED
   - MULTICS AVAILABLE, KVM AND KSOS 1980

6. CAPABILITY-BASED SECURITY
   - TRUSTED, MULTI-LEVEL OS
   - SECURITY POLICY ENFORCED BY HARDWARE "TAGS", AND SOFTWARE HIERARCHY
   - FUTURE DEVELOPMENT

System Development Corporation

PREDICTABLE IMPACTS BY 1985

1. INSTITUTIONALIZATION OF COMPSEC TECHNOLOGY
   * INCREASING & KNOWLEDGEABLE MARKET DEMAND
   * INTEGRATED COMPSEC REQUIREMENTS
   * FOUNDATIONS OF A PRODUCT APPROVAL MECHANISM

2. BURGEONING MARKET FOR TRUSTED PRODUCTS AND APPLICATIONS
   * STIMULATED BY AVAILABLE
     - SECURE OS (KSOS/KVM)
     - VERIFICATION TOOLS (SPECIAL, INA JO, GYPSY)
   * SPECTRUM OF FEASIBLE TRUSTED COMPSEC PRODUCTS
   * MARKET RETARDANTS MAY LIMIT GROWTH

3. FOUNDATIONS OF A FORMAL SOFTWARE ENGINEERING METHODOLOGY
   * TRUSTED S/W METHODOLOGY R&D APPLICABLE TO GENERAL S/W COST & RELIABILITY
   * BASED ON SYSTEMATIC, RIGOROUS, MATHEMATICALLY FORMAL PROCESS
   * USES INTEGRATED DEVELOPMENT ENVIRONMENT WITH TOOLS TO ENFORCE METHOD COMPLIANCE

System Development Corporation

INCREASING & KNOWLEDGEABLE COMPSEC MARKET DEMAND

1. DOD DRAFTING AND RELEASING GOOD COMPSEC PROCUREMENTS
   * C^3I
     - KSOS, PSOS, OASIS, WWMCCS, ICCS, KAIS
   * NETS
     - SACDIN, AUTODIN II, ENCRYPTION PROGRAMS, PLI, BCR
   * SPACE
     - SCF UPGRADE, SPACE SHUTTLE, SPADOC
   * LOGISTICS
     - MAC ROC 5-76
2. FED AND INDUSTRY BUYING
   * NBS
     - RISK ASSESSMENT (R/A) STANDARDS
     GSA
     - ENCRYPTION STANDARDS (DES, 1126, 1127)
   * SSA, HEW
     - FRAUD DETECTION
     - PRIVACY PROTECTION
   * BANKING
     - FED RESERVE EXPERIMENT
     - SACC EFTS
     - BANK EFTS
   * INDUSTRY
     - R/A
     - ACCESS CONTROL (PIN BOX)
     - FRAUD DETECTION

System Development Corporation

INTEGRATED COMPUTER SECURITY REQUIREMENTS CAN BE SEGMENTED BY FUNCTION

1. DATA CAPTURE AND DISPLAY
   - SUBJECT-OBJECT IDENTIFICATION/LABELS
   - AUTHENTICATION AND AUTHORIZATION
   - PHYSICAL ACCESS CONTROL

2. DATA TRANSMISSION
   - MSG-BASED TRAFFIC WITH CONTROL AND TEXT FIELDS
   - ERROR AND TAMPERING DETECTION PROTOCOLS
   - ENCRYPTION OF MSG TEXT END-TO-END
   - AUTOMATIC ENCRYPTION KEY MANAGEMENT

3. DATA STORAGE AND RETRIEVAL
   - DATA SECURITY LABELS
   - ITEM-LEVEL GRANULARITY
   - ACCESS CONTROL AND LOGGING — OS AND DMS
   - REASONABLENESS ENFORCEMENT (SEMANTIC, LIMITS, TIME)

System Development Corporation

INTEGRATED COMPUTER SECURITY REQUIREMENTS CAN BE SEGMENTED BY FUNCTION (CONT'D)

4. DATA PROCESSING AND CONTROL
   - DEDICATED USE-NO OPERATIONS AND DEVELOPMENT
   - MIX SENSITIVE DATA ONLY ON "TRUSTED" SYSTEM
   - ACCESS AND AUDIT CONTROL MECHANISM (ACM)
   - ACM MUST PROVIDE SELF PROTECTION FOR TRUST
   - ISSO DMS
   - APPLICATIONS MUST SUPPORT HIGHER LEVEL SECURITY PROTOCOLS

5. FACILITY AND OPERATIONS
   - PHYSICAL PERIMETER/EQUIPMENT ACCESS CONTROL
   - TRUSTED PERSONNEL AND PROCEDURES, ISSO
   - HW AND SW CONFIGURATION CONTROL
   - ISSO MANAGED DATABASE OF SECURITY PROFILES
   - ENFORCED OWNER REVIEW OF AUDIT LOGS

System Development Corporation

FOUNDATIONS OF A PRODUCT APPROVAL MECHANISM

1. CURRENT DOD POLICY ADMITS MLS ADPS

2. DESIGNATED APPROVING AUTHORITY (DAA) ACCREDITS EACH ADPS

3. DAA'S SUPPORTED BY COMPSEC TECHNICAL ASSESSMENT CERTIFICATION, A MAJOR FOCUS OF DOD COMPSEC INITIATIVE

4. CERTIFICATION BASED ON CREDIBLE EVIDENCE OF ADPS TRUST, EVIDENCE COLLECTED OVER SYSTEM LIFE CYCLE

5. EVIDENCE CREDIBILITY ENHANCED BY
   - SERIOUSLY WRITTEN PROCUREMENT RFP WITH COMPSEC-BASED EVALUATION CRITERIA
   - KNOWLEDGEABLE PROPOSAL USING COMPSEC DEVELOPMENT METHODS
   - COMPSEC DEVELOPMENT METHODS BASED ON FORMAL H/W & S/W ENGINEERING
   - SECURE SYSTEM OPERATION

6. APPROVED PRODUCTS LIST
   - MARKET SELECTS FROM ALREADY APPROVED PRODUCTS
   - MANUFACTURER BUILDS & SUBMITS PRODUCT FOR DAA CERTIFICATION

System Development Corporation

SECURITY ASSURANCE LIFE CYCLE

System Development Corporation

-7

SYSTEM LIFE CYCLE SECURITY OPPORTUNITIES

LIFE CYCLE PHASES

- CONCEPT DEVELOPMENT
- REQUIREMENTS DEFINITION
- SYSTEM RFP
- SYSTEM DEVELOPMENT
- SYSTEM IOC
- SYSTEM FOC
- ENHANCEMENT

OPPORTUNITY

- DO THREAT/RISK/COUNTERMEASURE ASSESSMENT
- CONSIDER MAJOR COMMITMENT TO A SECURE ARCHITECTURE TO PERMIT CONTROLLED SHARING, TRUSTED SEGREGATION
- INCLUDE MULTI-LEVEL SECURITY TO SUPPORT SHARING, PROTECTION
- REQUIRE SERIOUS SECURITY CONTROLS IN HARDWARE, OS, DMS, TP, TSS, NETWORKS
- STATE SECURITY THREATS AS FORMAL POLICY MODEL
- REQUIRE FORMAL TOP LEVEL SECURITY SPECIFICATION WITH CORRESPONDENCE PROOF
- REQUIRE SECURITY DRIVEN DESIGN

System Development Corporation

SYSTEM LIFE CYCLE SECURITY OPPORTUNITIES (CONT'D)

OPPORTUNITY

LIFE CYCLE PHASES

- CONCEPT DEVELOPMENT
- REQUIREMENTS DEFINITION
- SYSTEM RFP
- SYSTEM DEVELOPMENT
- SYSTEM IOC
- SYSTEM FOC
- ENHANCEMENT

- EMPLOY SECURITY ENGINEERING FOR ADP SECURITY
- EMPLOY SPECIAL SOFTWARE METHODOLOGY TO PRODUCE TRUSTED SOFTWARE
- REVIEW VENDOR PROTECTION FEATURES
- CONSIDER SECURE SUBSYSTEMS
- REVIEW THREAT/RISK/COUNTERMEASURES
- PERFORM PENETRATION ANALYSIS
- REVIEW COMPLIANCE
- CONSIDER SECURE SUBSYSTEMS AND SECURE DISTRIBUTED SUBSYSTEMS
- REVIEW COMPLIANCE, CERTIFICATION, AND RECERTIFICATION
- ENFORCE SECURE CONFIGURATION CONTROL

System Development Corporation

PREDICTABLE IMPACTS BY 1985

1. INSTITUTIONALIZATION OF COMPSEC TECHNOLOGY
   * INCREASING & KNOWLEDGEABLE MARKET DEMAND
   * INTEGRATED COMPSEC REQUIREMENTS
   * FOUNDATIONS OF A PRODUCT APPROVAL MECHANISM

2. BURGEONING MARKET FOR TRUSTED PRODUCTS AND APPLICATIONS
   * STIMULATED BY AVAILABLE
     - SECURE OS (KSOS/KVM)
     - VERIFICATION TOOLS (SPECIAL, INA JO, GYPSY)
   * SPECTRUM OF FEASIBLE TRUSTED COMPSEC PRODUCTS
   * MARKET RETARDANTS MAY LIMIT GROWTH

3. FOUNDATIONS OF A FORMAL SOFTWARE ENGINEERING METHODOLOGY
   * TRUSTED S/W METHODOLOGY R&D APPLICABLE TO GENERAL S/W COST & RELIABILITY
   * BASED ON SYSTEMATIC, RIGOROUS, MATHEMATICALLY FORMAL PROCESS
   * USES INTEGRATED DEVELOPMENT ENVIRONMENT WITH TOOLS TO ENFORCE METHOD COMPLIANCE

System Development Corporation

SPECTRUM OF FEASIBLE TRUSTED COMPSEC PRODUCTS

1. KSOS/KVM ENHANCEMENTS

   * PERFORMANCE TUNING & IMPROVEMENTS (E.G., KVM 1.5)

   * FUNCTION ADDITIONS/CHANGES (E.G., KVM 2)

   * NEW HARDWARE BASE (E.G., KSOS/SCOMP)

2. TRUSTED STANDALONE PRODUCTS (NO KERNEL OS)

   * CRYPTO DEVICES
     - LINE MULTIPLEXING
     - MSG MUX/SWITCH
     - MSG FIELD (E.G., PIN, $, CRC) ENCRYPTION
     - MSG GATEWAY (E.G., PIN REENCRYPTOR)
     - END TO END ENCRYPTION

   * MLS TERMINAL
     - ENCRYPTION CONTROL (I.E., KEYS, MSG FIELDS, PROTOCOLS)
     - TRUSTED DISPLAY/COMMANDS (E.G., RELEASE APPROVAL)
     - CONCURRENT LEVELS (E.G., SPLIT SCREEN & CHAR STREAM)
     - LOCAL ID AUTHENTICATION
     - LOCAL DATA TYPE ENFORCEMENT (E.G., LIMITS, LABELS, LOGIC)

   * JOB STREAM SEPARATOR
     - MLS PROCESSOR CONTROLS LARGER P.P. RESOURCE

System Development Corporation

G-9

SPECTRUM OF FEASIBLE TRUSTED COMPSEC PRODUCTS (CONT'D)

3. APPLICATIONS EXTENSIONS (KSOS/KVM)
- S/W UTILITIES
  - CLEAR MEMORY
  - LINK/LOADER
  - EDITOR
  - TRANSLATORS
  - DBUG
  - TEXT FORMATTING
- SYSTEM ADMINISTRATION
  - LOGIN
  - AUDIT
  - ACCOUNTING
  - SECURITY PROFILE MAINTENANCE
  - RESTART & RECOVERY
- SECURE DMS
  - MLS OBJECTS
  - FINE GRANULARITY
  - USER VIEW
  - ELECTRONIC MAIL/MSG
  - DATA TYPE CHECKER/ENFORCER
- SECURE NET DEVICES
  - SNFE
  - KDC
  - GATEWAY
  - MUX/CONCENTRATOR
  - SWITCH
  - SANITIZER
- TRUSTED S/W LIBRARIES
  - APPLICATION ALGORITHMS
  - USER INTERFACES

System Development Corporation

MARKET RETARDANTS MAY LIMIT GROWTH

1. TRUSTWORTHY DEVELOPMENT ENVIRONMENT
   - MASTER COPY CONTROL
   - CONFIGURATION CONTROL
   - LIFE CYCLE MAINTENANCE
   - METHODS & TOOLS

2. TRUSTED COPY DISTRIBUTION
   - TAMPER PROOFED COPIES
   - ROM
   - ENCRYPTION
   - CRC/ECC SCHEMES

3. INDUSTRY VS GOVERNMENT CONTROL
   - CLASSIFICATION
   - APPROVAL METHODS/PRODUCTS
   - STANDARDS
   - PRODUCTION ECONOMICS

System Development Corporation

PREDICTABLE IMPACTS BY 1985

1. INSTITUTIONALIZATION OF COMPSEC TECHNOLOGY
   - INCREASING & KNOWLEDGEABLE MARKET DEMAND
   - INTEGRATED COMPSEC REQUIREMENTS
   - FOUNDATIONS OF A PRODUCT APPROVAL MECHANISM

2. BURGEONING MARKET FOR TRUSTED PRODUCTS AND APPLICATIONS
   - STIMULATED BY AVAILABLE
     - SECURE OS (KSOS/KVM)
     - VERIFICATION TOOLS (SPECIAL, INA JO, GYPSY)
   - SPECTRUM OF FEASIBLE TRUSTED COMPSEC PRODUCTS
   - MARKET RETARDANTS MAY LIMIT GROWTH

➤ 3. FOUNDATIONS OF A FORMAL SOFTWARE ENGINEERING METHODOLOGY
   - TRUSTED S/W METHODOLOGY R&D APPLICABLE TO GENERAL S/W COST & RELIABILITY
   - BASED ON SYSTEMATIC, RIGOROUS, MATHEMATICALLY FORMAL PROCESS
   - USES INTEGRATED DEVELOPMENT ENVIRONMENT WITH TOOLS TO ENFORCE METHOD COMPLIANCE

System Development Corporation

SYSTEMATIC, RIGOROUS, MATHEMATICALLY FORMAL PROCESS

1. PROCESS STEPS FOLLOW IMPLICATION CHAIN
   - CODE → HOL → SPEC → MODEL → POLICY
   - CORRESPONDENCE (I.E., →) VALIDATED BY VERIFICATION PROOF

2. STEPS FORMALLY STATED IN PRECISE LANGUAGE
   - POLICY
     - DIRECTIVE 5200.28 IS DOD STANDARD
   - MODEL (TLS)
     - FORMALIZE ACCESS CONTROL POLICY AS CORRECTNESS CRITERIA (INVARIANTS)
     - SUBJECT PROCESSES, MLS OBJECTS
     - INITIAL STATE DESCRIPTION
     - ALLOWABLE (SECURE) STATE TRANSITIONS
     - CORRESPONDENCE TO POLICY BY INFORMAL REVIEW
   - SPEC
     - FORMALLY STATED IN RIGOROUS PREDICATE CALCULUS, NON-PROCEDURAL LANGUAGES (E.G., SPECIAL, INA JO, GYPSY)
     - TOP AND REFINED LEVELS OF ABSTRACT SPECS
     - VERIFY SECURITY CORRECT BEHAVIOR SPEC TO MODEL
     - SUCCESS AT MITRE, SDC, I.P. SHARP, FACC
   - HOL
     - SECURITY-RELEVANT PARTS (E.G., KERNEL) OF SYSTEM
     - PROCEDURAL LANGUAGE AMENABLE TO VERIFICATION (PASCAL, MODULA, EUCLID, GYPSY, ADA)
     - HOL-SPEC MAPPING VERIFIED
     - EXPERIENCE LIMITED BUT INCREASING WITH TOOL MATURITY
   - CODE & H/W
     - ACTUAL EXECUTING PROCESSES
     - VERY LITTLE VERIFICATION ACTIVITY TO DATE · HARD AND COSTLY R&D
     - COMPILER AND S/W TESTING FOR CORRESPONDENCE

System Development Corporation

G-11

TOOL-ENFORCEMENT KEY TO METHODOLOGY SUCCESS

1. A NUMBER OF METHODS FDM (SDC), HDM (SRI), GYPSY (U OF T), AFFIRM (USC ISI), PDS (HARVARD), ...

2. ALL FOLLOW TOP-DOWN APPROACH OF STEPWISE DESIGN *REFINEMENT*

3. SPEC/HOL MODULES ANALYZED BY TOOLS FOR SECURITY CONDITIONS YIELDING PROPERTIES/ASSERTIONS TO BE PROVED (THEOREMS)
   - STATE VARIABLES LEGALLY (SECURELY) SET/USED
   - STATE TRANSITIONS RESULT IN SECURE END STATE
   - SPEC PROCESSOR OR HOL VCG TOOLS

4. THEOREM PROVERS VERY EFFECTIVE
   - AUTOMATIC AND INTERACTIVE TPs IN ACTIVE USE
   - PROOFS LONG & DETAILED, BUT NOT DEEP
   - TP MECHANIZES PROOF BOOKKEEPING, AVOIDS MISTAKES, OVERSIGHTS
   - TP FORMATS HUMAN READABLE STEPS TO PROOF, OR TO POINTS OF FAILURE
   - PROOF/FAILURE INTERACTIVE DESIGN PROCESS

5. OTHER TOOLS
   - FLOW ANALYZER
     - EXAMINES SOURCE CODE DATA FLOW CONSISTENT WITH SECURITY LEVEL
   - CONFIGURATION CONTROL
     - MAINTAINS SPEC/HOL SOURCE FILES *STATUS AND DEPENDENCIES* LINKED FOR (RE) PROOF
   - TEST CASE GENERATORS
     - USES HOL ASSERTIONS TO AID IN TESTING CODE
   - DOCUMENT CONTROL
     - SOURCE TEXT, PROOF, ENGLISH DESCRIPTIONS

PREDICTED IMPACTS ARE NOT SPECULATIVE

1. INSTITUTIONALIZATION NOW IN PROGRESS
   - NBS, OMB, GSA, SSA, ...
   - DOD SECURITY INITIATIVE/CONSORTIUM
   - GOVERNMENT REGULATIONS
   - INDUSTRY PROCUREMENT/INVESTMENTS

2. MARKET INCREASING
   - A DOZEN OR MORE PROGRAMS IN PROGRESS AT SDC

3. FORMAL DEVELOPMENT METHODS ARE WORKING
   - EXPERIENCE MOUNTING IN S/W RELIABILITY AND REDUCED TESTING
   - IMPROVED DOCUMENTATION OF DESIGN
   - SUPERIOR PROGRESS REVIEWS BASED ON *RIGOROUS SPECS* AND PROOF EVIDENCE OF PROGRESS
   - TOOLS ARE WORKING AND BECOMING AVAILABLE

4. SECURITY PROGRAM STIMULATING MORE COMPSEC R&D
   - NEW COMPUTER ARCHITECTURE (PSOS)
   - DISTRIBUTED PROCESSING
   - BROADER POLICIES/MODELS
     - RELIABILITY
     - PERFORMANCE
     - DENIAL SERVICE
     - PROTOCOLS
   - S/W ENGINEERING METHODS & TOOLS

G-12

"Computer Security Impacts on Future
System Architecture"

Mr. Edmund Burke
The MITRE Corporation

Computer Security Technology

Future Directions, Future Needs

Outline

- **COMPUTER SYSTEM TECHNOLOGY**
    - • HARDWARE AND SOFTWARE DIRECTIONS
    - • FUTURE SYSTEM ARCHITECTURES

- **COMPUTER SECURITY DIRECTIONS**
    - • COMPUTERS
    - • NETWORKS

- **NEEDED TECHNICAL STIMULI**

Computer Hardware

- GROWTH IS EXPLOSIVE

- COST/PERFORMANCE DROPPING
  BY ORDERS OF MAGNITUDE

- VLSI PROMISES-SMALLER, FASTER, CHEAPER

Computer Software

- SOFTWARE DEVELOPMENT STILL AN ART
  - CURRENT TECHNIQUES ARE CODIFIED COMMON SENSE

- SOUND ENGINEERING BASIS STILL SOUGHT

- ACADEMIC COMMUNITY PURSUING FORMAL TECHNIQUES

H-2

Distributed Capabilities
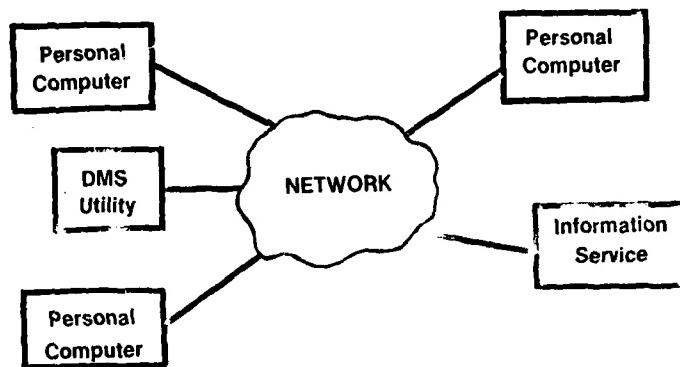
Emerging Systems Architecture

- PERSONAL COMPUTERS

- LARGE SCALE UTILITIES

- COMMON CARRIERS FOR COMPUTERIZED TRAFFIC

Personal Computers

- O/S AND APPLICATIONS "TUNED" TO A SINGLE USER

    PET, TRS-80 — TOO SMALL

    OS, MULTICS — TOO BIG

    UNIX $^{®}$ — ABOUT RIGHT

® UNIX IS A TRADE/SERVICE MARK OF THE BELL SYSTEM

Personal Computers

APPLICATIONS

    ELECTRONIC MAIL

    WORD PROCESSING

    PERSONAL FILES

    AGENT FOR ACCESS TO COMPUTER UTILITY

Large Computer Utilities

- INFORMATION SERVICES

- DATA MANAGEMENT SYSTEMS

- COMPUTATIONAL CAPABILITIES

Current State of Computer Security

- SECURE SOFTWARE SYSTEMS EMERGING

    - MANUFACTURERS BEGINNING TO MARKET PRODUCTS

- INTEGRATION OF COMMUNICATIONS AND COMPUTER SECURITY NEEDED

    - NO COMMON CARRIER OFFERS MUCH

- SOFTWARE ENGINEERING (& VERIFICATION) TECHNOLOGY

H-5

Scenario For The Future
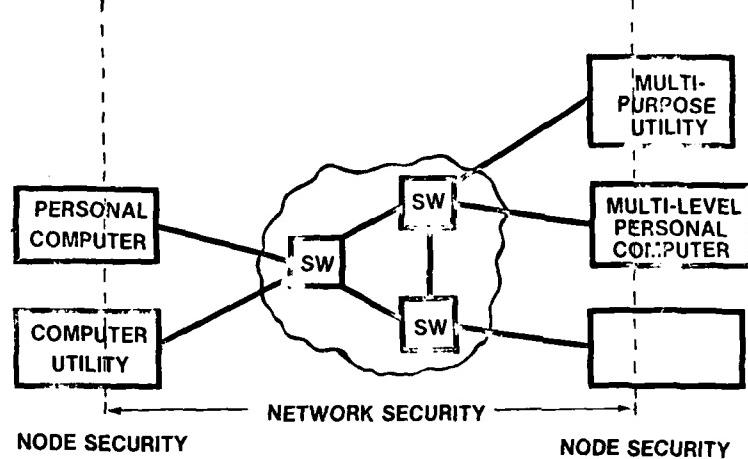
Needed Developments

- INTEGRATED COMMUNICATIONS AND COMPUTER SECURITY

  - SECURITY FEATURES AVAILABLE FROM COMMON CARRIERS

  - WIDER RANGE OF SECURITY EFFECTIVENESS FROM MANUFACTURERS

H-6

Needed Developments

- UNIFORM ACCESS CONTROL POLICY

    - CONSISTENT SET OF SENSITIVITY LEVELS

    - PROVISION FOR ORGANIZATIONAL PREROGATIVES

    - LEGAL STRUCTURE

Needed Developments

- FURTHER DEVELOPMENT OF FORMAL ENGINEERING DISIPLINES

    - DESIGN AND IMPLEMENTATION VERIFICATION OF
      CRITICAL SOFTWARE COMPONENTS

    - EXTENSION TO CRITICAL HARDWARE COMPONENTS

H-7

Summary

- NEW HARDWARE DEVELOPMENTS CHANGING SYSTEM ARCHITECTURES

- CHEAPER DATA COMMUNICATIONS PERMITTING DISTRIBUTION OF COMPUTERS

- INTEGRATED SECURITY CONTROLS NEEDED FOR HETEROGENEOUS HOSTS ON INTERCONNECTED NETWORKS

- FORMALISMS NEEDED TO PROVIDE ASSURANCE OF SYSTEM SECURITY

WHAT EVERY VENDOR ALWAYS WANTED TO KNOW ABOUT
GOVERNMENT COMPUTER USERS' SECURITY NEEDS
(but was afraid to ask)

Dr. Ted M. P. Lee, Sperry-Univac
Jim Anderson, James P. Anderson, Inc

[This is written as a questionnaire to be answered by a suitably representative sample of government computer customers. The focus is mostly on future systems wherein a need for true multi-level security might appear, but it is also intended to elicit an indication of the current state of affairs wherein security problems are wishfully ignored, limited, or avoided by administrative, personnel, or physical security measures. It is recognized that most installations are a mix of applications and problems, so that a single answer will generally not suffice. Note that the questions will for the most part also apply to non-government users. The questions have been obtained from an informal canvassing of several vendors by T. M. P. Lee.]

[The answers have been prepared by J. P. Anderson and are an attempt at an objective look at the whole issue of computer security and how it is handled in DoD and other parts of the Government.]

[Editor's note: the answers represent the views of J. P. Anderson and do not necessarily represent the views of the Department of Defense or of the U. S. Government.]

1. Customer Background

    1.1: Awareness

        Does the customer know what he's talking about?

    Answer: It depends. In some Government units, there is
             considerable knowledge, and much of that is available to
             the managment of those units (e.g. Intelligence
             Community Agencies, some parts of DoD, etc.) In other
             parts of Government activities (e.g. the civilian
             agencies), there is only the vaguest appreciation of the
             problem, and little of the solutions.

    1.2: Point of Contact

        Who is the right person at each agency, department, division,
        etc. to ask these questions of?

    Answer: There is no SINGLE point of contact in each Agency. You
             will get a better perspective on the needs by talking
             with the Data Processing people than with the security
             people. However, the vendors must talk to the ultimate
             customer in order to fully appreciate the requirements.
             In many agencies, the Data Processing people will buy and
             operate computers for an operating branch/activity with
             the substantive application. In these cases, DP is an
             'agent' for the ultimate customer.

I-1

1.3: Decision Maker

Who makes the purchasing decisions?

What does he know or care about trusted computer systems?

Answer: In general, some form of official or unofficial committee. The committee members in general have little or no great concern for/about trusted computer systems.

1.4: Purchasing Criteria

a) What criteria are used for purchase decisions?

b) Are there any written standards (viz-a-viz security)?

c) Where do the criteria come from -- internal? user groups? law/regulation? gut feel?

d) How firm and precise are the criteria? Can I see them? How will they change?

Answer: a) and b) It depends on the agency, however, compatibility with existing applications/software, other hardware units etc. is often the key criteria. Security 'standards' exist in part, but in many cases, they are a recitation of 'nice features' rather than a functional set of security requirements based on the intended use of a system.

As an expample, it might be reasonable for an agency to specify that a computer has sufficient mechanism to isolate/control transaction users, to include at a minimum a user-identification/authentication mechanism, and system software to utilize the mechanism, and to mantain it. The specification could reasonably describe how the resultant system is expected to be used; that the system should have sufficient internal mechanism to support the building of a 'secure' transaction system, etc. The specification may also state that the system does NOT have to provide control against 'malicious programmers', since all of the customers programmers are or will be cleared. (It is a gross understatement that such specifications are not commonplace)

c) The purchasing criteria (security and otherwise) come first from internal sources (e.g. DP shop)), leavened by regulations that everyone is generally aware of (e.g. DOD 5200.28, DCID 1/16, etc.). There is little or no 'gut feel' .

d) The criteria are very firm and NOT very precise. In virtually every case, they can be seen. The regulations undergo more or less continuous change. the changes are SLOW, and evolutionary because of the inertia that has to be overcome in order to effect change. The source of the change is often economic.

In order to reduce cost or for some other economic reason, a proposal will be made to relax or modify a security 'rule'. This is then debated, sometimes endlessly and without resolution, but occasionally a change will emerge.

The regulations will evolve as it is possible to

(1) Demonstrate that a mode of processing hitherto thought 'impossible' can be successfully supported with only minimum risk etc.

(2) Show that the economics for making a change are favorable.

1.5: Intangibles

What intangible factors play a role in purchase decisions? -- e.g., newness for newness' sake, keeping up with the Joneses, gee-whiz technology?

Answer: There is nothing I can comment on regarding the intangibles. I would say that the intangibles regarding a vendor, and how he is perceived by a particular customer are VERY MUCH more important than any security questions and/or the like.

2. Data Processing Environment

2.1: Use

a) What does the customer do with his computers?

b) What is the use of the systems by percentage, i.e.:

> communications systems
> data processing systems
> embedded control systems?

Answer: a) Everything. More and more users are interfacing to computers as transaction users.

b) What kind of systems? If the question is what is the expected use of trusted systems (by percentages), then as a guess:

DP	50 - 70%
Commo	30 - 20%
Control	20 - 10%

2.2: Configurations

What kind of configurations are involved -- large/small? centralized/distributed? networks?

Who are his current vendors?

Answer: Everything, but:

        a) More networks

        b) From a), more distributed

        c) Large and small

2.3: Applications

What spectrum of applications are involved: query, limited function subsystems, data management, full-scale user programming?

Answer: As noted above, there is more fully developed systems (applications) in place, mostly based on interactive transactions by users. Batch is still used either as a hangover, or as the method of choice for such applications as payroll, etc. Even with batch systems, there is a some evidence that networks are being used to collect files, and disburse the results.

Full scale user programming is less frequent than was the case 10 years or so ago. Except in some 'scientific' shops, most programming is done in support of the development and or maintenence of transactional/ interactive applications, where the bulk of the use of a system is concentrated.

2.4: Security Severity

What mix of data sensitivity and personnel trustworthiness would the customer like to be able to support?

Can a clearance/classification matrix be given?

Answer: A VERY Broad brush treatment of the topic..... NOT GOVT POLICY! (But a guess at what would satisfy if it were really available now).

Civil-Agencies: Privacy Act data, Agency proprietary data, and Uncleared people.

DoD-Low: Secret Confidential and Unclassified Data with users at all three levels simultaneously ('true' multilevel, low grade)

DoD-Medium: Top Secret, Secret data, with users cleared at Top Secret and Secret levels ('true' multilevel -- medium) (like AFDSC)

DoD-High: Top Secret, SCI, with users at TS (only) and TS(SCI) levels

Intel.-Comm.: Top Secret (SCI), TS, Secret, Confidential, Unclassified with users at TS(SCI) levels. (Need to Know and Proprietary information protection also required)

I-4

3. User Security Policy

   3.1: Perception

      a) What does "security" mean to the customer?

      b) Does he care enough about the problem to want the very best, or will #2 be good enough?

   Answer: a) Varies with the type of customer. With most, it is secondary or tertiary consideration to questions of efficiency, functionality, compatibility and the like. Most customers DO NOT have a clear, thought-out notion of where security fits, or how much emphasis to give it. To most, security means badges and access controls at the entrance to computer rooms. To the extent that a threat is perceived, it is seen as an external threat.

   b) No.

   3.2: Importance

      What is the relative (and ""absolute", if you can determine it) importance of the three faces of security -- integrity, availability, and confidentiality?

   Answer:

	Relative	Absolute
Integrity	2	2
Availability	1	1
Confidentiality	3	3

   3.3: Threats

      a) What are the perceived threats to the aspects of security mentioned above and what is their relative importance (or seriousness)?

      b) Does the customer know or care about the malicious programmer threat?

      c) About covert channels and Trojan Horses?

      d) About subversion in the vendor's development and maintenance processes?

Answer: a) EXTERNAL— MOST organizations are focused on the external threat to the exclusion of all others. They are incapable of thinking that one of their 'own' could be a black hat. Even when they choose to to think of it, their activities are based in the most part on an external threat except for the relatively simple physical/procedural aspects (controlling access to the computer center, etc.)

b) Because of the growing use of transactional systems, where users are NOT programmers, most people do NOT see the malicious programmer as a serious threat. There are very few places where 'general use' programming of systems is supported or needed by the operational arms of the agency (payroll, personnel, etc. etc.). Programmers, where needed are cleared to system high, and are not especially controlled.

c) Huh? Most people do not even acknowledge b). Channels are 'academic' finds. They are theoretically possible, but are not readily understood by the laiety partly because they are NOT considered an important threat since no one today would clear the receiver-agent high enough to get access to the transmitter, or the receiver-agent would already have full access from being a programmer or some such. This is not to say that the channels are not important, it is just that they defy general solution today.

d) See b) and c). NO. In general, do not see the threat as 'real'. It is in the same class as an airplane dropping onto them. Possible, but not very likely.

3.4: Policy

Does the customer have a security policy that applies to his ADP operations?

Is it written down? Followed and enforced?

Answer: In most cases, yes; DOD 5200.28, or AF=====, or etc. These policies are followed in the main because they deal mostly with tangible things: locks, badges, checklists, etc.

3.5: Access Criteria

Assuming Harry Smith (or Ivan Ivanovitch) asks to (read write, execute, ...) (file, program, record, ...) XYZ, what criteria would the customer like to use to grant or deny the request? (security labels, privacy requirements, "need-to-know" -- can anyone say what that means? -- access lists, ...)

Answer: It doesn't really work that way. First, Harry (Ivan) is/has to be employed by an Agency/ Contractor etc. His JOB must require use of a computer. His BOSS must approve (pay for) an account, etc. By the time he gets around to asking to Read, Write, Execute and so forth, he is already known to the organization, DP shop, etc. His access rights are derived from a) his job, b) his 'clearances' . After that, ALL methods of granting and controlling accesss are used; security levels (infrequently) privacy requirements (not very common in my experience. There is some, but not much), Need to Know (often involved, but rarely labeled), access lists (proably the most common). Access lists are becoming the wave of the future, ORCON as the ultimate in control.

3.6: Granularity

Down to what level of granularity of data (e.g., file, record, field) is it necessary or desirable to enforce the policy?

Answer: (Yes)**3

3.7: Role of System

a) To what extent ought or must the ADP system (operating system) make or enforce the decision to grant an access request, or to what extent can that be handled outside the system without excessively limiting its usefulness?

b) When do you expect to have enough confidence in the "security" system kernel that users may "turn off" other protective measures?

Answer: a) Presently, systems are only marginally involved in enforcing access rights. They should be substantially involved in the future.

b) Perhaps never. I don't believe that it was EVER proposed that security kernel technology was an exclusive solution to the problem. One would still expect to have passwords to control access to the system, even with the SKT, one might expect passwords to control discretionary access (redundantly perhaps), Marking of files/output etc, while not a control, is a form of 'protective' measure that should/might be retained regardless of whether the system operated under an SKT or otherwise. The question assumes (as did several other questions in this series) that security is a tangible add-on rather than an integrated design approach.

3.8:   Audit

What security audit records would the customer like to have
kept?

What does he keep?

Is he prepared to process them?

(How would he like to process them?)

Answer: None special. Most capture illegal log-on attempts.
          Most SECURITY audit data is weak or not very useful.
          Most users KEEP none of the audit data, except a few
          places that pile up the operators and audit trail logs
          "in case" they ever have to do a damage assessment. To
          be really useful the processing should result in
          exception reports. Six sice inches of listings are not
          very useful as audit data.

3.9:   Special Requirements

What special technical requirements does the customer have
that are not covered so far? (e.g., TEMPEST, marking,
specific human interface protocols.)

Answer: TEMPEST. Most of the others are left out by and large.

4.   Technical Security Policy

4.1:   Certification

a) What does "certification" mean?   .

b) Will there ever be an institutionalized process with clear
and visible standards of acceptance? When? Where?

c) Who will have the final authority for the "certificate" of
certification of the secure system software?

d) How will the DoD go about certifying a system os as to be
considered "trusted"?

Answer: a) As a personal observation, it will mean that if
          security is involved in the ultimate application of the
          system, that buyers will have a hard time justifying the
          buying of uncertified systems.

          b) Yes, by 1985 (+/-) ; in DoD and Intelligence
          communities, not likely in Civilian agencies by then
          because of the relatively short attention span of
          Congress, and others who started the Privacy Act moves.

          c) The buyer. He is the FINAL authority on anything.
          even today, the individual system operator in the USAF
          could make his own independent judgment to run
          multi-level WITHOUT any further blessings or approval.

          d) Pass.

4.2: Assurance Measures

a) What tests/measures/evaluation criteria is the customer
using, or would like to use, or will use, to assure himself
that all the technical security enforcement measures in his
system work as they are supposed to, in the face of the
perceived threats?

b) How much does he want to monitor and be involved with the
vendor's design, development, and support procedures?

c) Who will maintain the security SW kernel? Since DoD
maintains/or pays to maintain significant amounts of SW for
embedded computer systems and other stand-alone systems, why
shouldn't the DoD plan to do the same for a special security
package?

Answer: a) None -- Working with the system

b) None. How much do most people want to be involved in
the design and production of automobiles (altho since the
Chrysler bail-out, maybe the will of the country is that
everyone one wants such invlovment). In general, one
wants someone (e.g. an FTC) making sure that the autos
are safe at some speed, but NOT designing everthing into
them.

c) Could be anyone. Maybe a software house such as
CSC/SDC etc. could/should be the one.

4.3: Standardization

a) Is there any hope for a common direction to emerge across
a "suitably" large segment of the market place? -- i.e., is
there reason to expect the DoD, Intelligence Community,
Federal Government as a whole, State and Local Government,
Private Sector, and Foreign Public and Private Sectors to
agree (sufficiently) on the nature and extent of the problem
and on acceptable solutions?

b) Will multi-level security start to become a mandatory
requirement in future RFPs where necessary?

c) Do you plan to specify trusted system requirements for the
next generation WWMCCS-(WIS)?

Answer: a) There is already a number of 'common' directions:
language standards, communications standards, etc.
HOWEVER, most of these standards have been devised or
heavily influenced by manufacturers in order to permit
competition (or thwart dominence of the market by one).
It is unlikely that all of the entities named will see
much in common because of their responsibilities AND
perceptions differ so widely.

b) Yes

c) Pass (I would so plan, but then, I am NOT WWMCCS etc.)
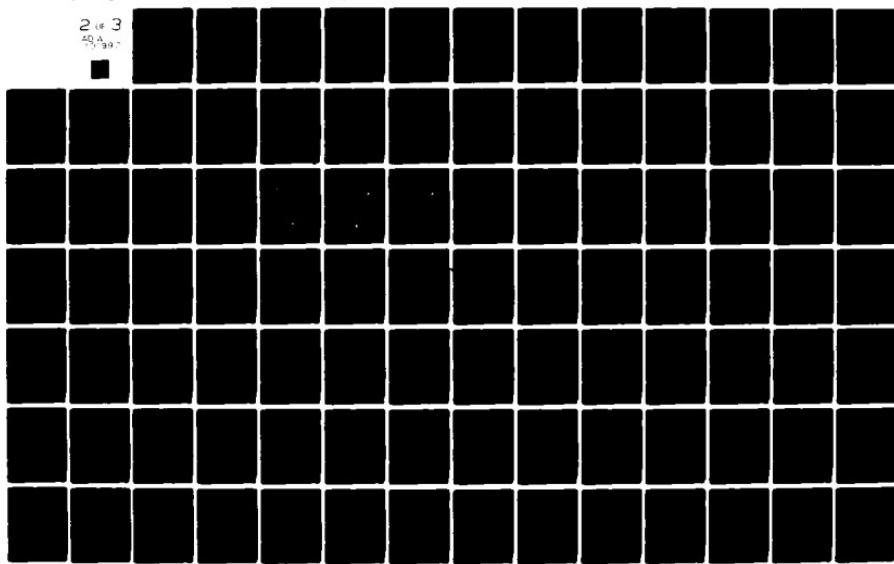
4.4: Technology

a) Do you have strong biases for one kind of security technology (mechanisms or architecture, hardware or software, development and verification procedures) versus another? Why?

b) With the availability of the 32-bit super minis, will a requirement for 16-bit secure minis still be necessary?

c) What is the government position on software vs. firmware security "fixes"?

d) Do you plan to use ADA as the language for the security kernel? If not - why not?

e) Will a computer system require "special features" in order to use the security kernel, i.e., something not in a manufacturer's standard product line?

f) Do you expect the use of the security kernel to require changes to the application SW packages; e.g., very little to significant?

Answer: a) Personal biases are: 1. For transaction systems-- they are common, and THE way commputers have gone and will continue to go in the future. 2. For 'distributed' (network) systems as models of architectures that should be built.

b) With the availability of 64 bit large machines, will a requirement for 32 bit secure minis still be necessary? The question indicates a naive belief that the word size is the key issue. The question should be: Is there now, and will there continue to be a market for secure minis? Answer: Yes.

c) Beats me. I personally oppose ANY KIND of 'fix'. Rather, I would prefer to see the systems used in environments where reduced user functionality (e.g. transaction systems) limits the risk.

d) It has been suggested as far as I know, but there is no special (pun NOT intended) reason to do so. The question is somewhat irrelevant.

e) 'Special features' are the hall mark of basically a single manufacturer. Most others design into their systems the basic structures needed for the desired functionality, then implement the design in hardware, firmware, or software depending on the performance or other needs. In order to 'use' (implement) a security kernel a machine must aprehend in some fashion the concept of 'process'. In order to do this, it may use such hardware as 'descriptors', mapping registers, etc. Basically the hardware is used to provide efficient enforcement of access decisions (i.e. policy decisions) made by the operating system.

f) For some packages the introduction of a security kernel will require significant changes to the package. For example, it is possible that the package (i.e. an application) itself may be multi-level. If this is so, then parts of it may require change (or at least partitioning) in order to isolate the multilevel parts. On the other hand, in other environments, it may be sufficient to group like users of an application, and have as many 'copies' of the application as the security levels require.

4.5: Classification

a) What aspects (how much) of a trusted system is going to need to be classified, to what level, and why? (inspection or alteration)

b) *How much of a trusted system needs to have been developed by cleared people in a cleared facility?*

Answer: a) None should be CLASSIFIED. Some systems may require the operational versions to be PROTECTED as classified (i.e. handled in trusted channels, etc.).

b) All of the 'trusted' parts. In most systems this is OK for the operating systems. There is still the trusted parts of applications built on trusted O.S. that need the protection of cleared people, etc.

4.6: Export

a) What classes of systems developed and approved for DoD (or other government) applications can be marketed and sold to other users?

b) Will there be any foreign export control problems?

Answer: a) None.

b) don't think we should export ANY reasonably high technology software or hardware. That is just shooting ourselves in the foot.

4.7: Credibility

a) Why should we pay any attention to "trusted" operating systems (especially the current R&D prototypes) when the underlying hardware and microcode are getting more complicated and hence less trustworthy?

b) Is the government not exposing a credibility gap by seeming to champion software security technology almost to the exclusion of hardware security problems and solutions? And by so far only producing "toy" or prototype systems?

c) *What is the status of end to end encryption efforts? If successful, do you see this method of achieving security pre-empting other efforts?*

Answer: a) *Hardware and microcode complexity is indeed a problem,
but one that can be attacked after the software problem
has been solved. (I guess that is to say that the
software problem looks easier at the outset). It is also
true that hardware and microcode complexity makes
manipulation possible by fewer people (who could be
controlled by other means?) even if the manipulation may
be practically undetectable.*

*b) There is not a credibility gap in terms of need. It
is true that most of the efforts have been directed to
software solutions. This was deliberate for several
reasons. First, it was believed (and still is) that the
software problem is 'easier', and more tractable.
Second. There is/and was research underway in highly
reliable systems that appeared to bear on the security
problem at the time. Finally, it was recognized that
'solutions' that might involve changes to hardware were
very unlikely to have any impact on the major
manufacturers since they were for the most part frozen in
their architectures. Therefore there isn't much room for
new designs.*

*The fact that the solutions have to date been to 'toy'
problems is in part a reflection of weak resolve. The
Multics project nearly made it.*

*c) Getting there. Even if it meets the most wildly
optimistic expectations, it will only complement other
efforts. Secure communications (even end-to-end) is NO
substitute for secure computers.*

5. Economics

   5.1: Value

   a) How much (assuming that can be given a reasonably precise
   meaning) security is the customer willing to pay for?

   b) How much is he willing to pay?

   c) How much of other things is he willing to sacrifice for
   security (e.g., performance, usability, integrated data
   bases, ...)

   d) Will government agencies tolerate the necessary
   degradation for certified security systems vs.
   non-kernelized systems?

   e) What impact do you expect to see (in terms of DP system
   degradation) when you use the security system kernel?

   f) Will the government pay the necessary delta for the
   security hardware needed in this type of system?

Answer: a) Not too much. The problem is that he gets along without it by spending different kinds of dollars (e.g. O&M) on different things that do not bear the same scrutiny as do Procurement dollars.

c) and d) 'Performance' might be eroded up to 20-25% in some settings without penalty. In most however, no detectable performance penalty would be permitted. If 'degradation' is greater than 50% then the vendor(s) have failed.

e) Varies with the expected use of the system. Present versions of the SKT might vary from 25 to 100% depending on the hardware upon which it is built, and the application for the system.

f) Only if it is the Nile -- it won't pay the Mississippi. This question is a reflection that security is separable from good design, can be priced, and added on like racing stripes and wire wheels. (Where is the Necessary River?)

5.2: Conversion

How willing is the customer to go through a (minimal, moderate, extreme, replacement) conversion (hardware, software, or operational procedures) to achieve better security?

Answer: Probably none at all. The challange will be to provide improved security in an 'invisible' (performance/user impact) way. Clearly not possible in the large, but a goal worth shooting for.

5.3: Business Forecast

a) Please forecast future procurements dependent on security -- $ worth of systems at security level of difficulty X (category X in the "evaluated products list"), mode of use Y, environment Z? (with specific procurements and dates)

b) Can you estimate the government market demand for secure systems for the next five years?

c) What will be the first "big buy" that will require MLS? What will be the time frame for the first MLS buy?

Answer: a) ????punt

b) At the end of 5 years (i.e. circa 1985 (+/-)), it is expected that trusted systems will be routinely required in all but single-function stand-alone systems. It is NOT expected that the Government will make a wholesale replacement of existing functioning systems. Once a secure system is seen to work, it will become 'standard'.

c) Ask your marketing people. Now. As soon as it is available.

6. Competition

    6.1: Questions

        What are you (the customer) asking of my competition?

    Answer: What is UNIVAC (Honeywell, Burroughs, NCR, IBM, DEC...etc...) doing?

    6.2: Answers

        What are they telling you?

    Answer: That they (Anyone, not YOUR company) are working on the problem, but we (the competition) have the solution.

    6.3: Guesses

        What do you (the customer) think they are going to do?

        Why should I believe what you tell me?

    Answer: Continue working on the problem, then follow IBM. That is what most manufacturers do.

THE DEPARTMENT OF DEFENSE COMPUTER SECURITY INITIATIVE PROGRAM

AND

CURRENT AND FUTURE COMPUTER SECURITY POLICIES

Presentation for the "Second Seminar on the Department of Defense
  Computer Security Initiative Program"

National Bureau of Standards, Gaithersburg, Maryland

January 16, 1980

Prepared by: Mr. Eugene V. Epperly
              Security Plans and Programs Directorate
              Office of the Deputy Under Secretary of Defense (Policy Review)

INTRODUCTION

I am delighted to participate in these seminars because the objectives
and efforts of the Computer Security Initiative Program are so closely
related in theory and practice to the security policy responsibilities
of the office I represent. Indeed, the initiative itself represents an
activity many of us realize is long overdue, and it has collectively the
potential to make a substantial contribution to a number of requirements
in today's world, well beyond the area of protecting computer processed
classified information.

It is both of these areas that I would like to explore today. As a
point of departure, let me start with a textbook definition of a policy
as simply a decision made in advance, that is, independent of a specific
instance or particular situation. A security policy would involve some
asset of value, some threat thereto, vulnerabilities and a resultant
risk scenario, and finally a decision concerning relative allocation of
resources for protection.

My primary focus will be upon current policy concepts and their frame-
work, to apply these to the objective of the Computer Security Initia-
tive and to apply these, in turn, to the broader environment established
by recent OMB (Office of Management and Budget) computer security poli-
cies.

BACKGROUND

DoD Security Policy Function

As was indicated, the office I represent is primarily concerned with security policy; specifically we function as principal Department of Defense advisor on matters of security policy, which in turn includes among other things, sensitive information, property and facilities of the department world-wide. Our office, moreover, is also the executive management agent for industrial security policy matters for sixteen other Executive Branch departments and agencies in addition to the Department of Defense, a program which encompasses over 11,000 industrial facilities in the private sector and involves over a million personnel security clearances.

What is common in our program with any effort to secure computer-resident information and related ADP assets is the need for a multi-disciplinary perspective using diverse talents, a systematic and comprehensive analytic approach and ultimately the identification and selection among these tradeoffs (Figure 1), involving generally: security, cost, effectiveness and efficiency factors.

Nonetheless, in the context of this presentation, the primary security function with which we deal involves the formulation and establishment of overall security policy for the protection of classified information; that is, Federal Government information and material which, because it bears directly on the effectiveness of our national defense and the conduct of our foreign relations, must be subject to some constraints and protection.

Problem First Surfaces

Interestingly, the problem of computer security was first formally surfaced in the Office of the Secretary of Defense through the DoD Industrial Security Program. In April 1967, a memorandum sent to our office expressed concern for the development of security policy and guidance for evaluating the security posture of computer systems, particularly those in a time-shared mode, and further stressed anticipation of a growing use of computers by defense contractors.

Because of the technical facets of the problem, we solicited the assistance of the Director of Defense Research and Engineering (DDR&E), who in June 1967 advised that the Advanced Research Projects Agency (ARPA) had been assigned responsibility: to identify the technical aspects of the security problems in time-sharing computer operations, to consider alternative solutions and to make recommendations for a preferred solution. Following discussions involving people from the university and industrial communities, a task force was formed in October 1967 consisting of a steering group, a policy panel and a technical panel. The Task Force was chaired by Dr. Willis Ware who addressed the previous of these seminars.

Nature of the Problem

The perceived nature of the problem impacting security policy at that time is best summarized by the following extracts from an internal office memorandum.

> Although the broad policy guidance of DoD Directive 5200.1 included adequate security guidance at this level for single-user ADP systems, it is inadequate insofar as the security needs posed by multi-level, time-sharing computer systems are concerned. Those time-sharing computer systems, in which many files of differing security classifications are processed simultaneously under the control of several terminal operators having differing security clearances and validated need-to-know, present a policy problem which is nowhere covered adequately by existing DoD Directives.

The Defense Science Board Task Force describes the problem in essentially the same way, in slightly different terms. The 'bottom line' was that remotely accessed resource-sharing systems introduced new complexities and issues, in turn not amenable to solution through the elementary safeguard of physical isolation [1].

The resultant Defense Science Board Report, entitled, "Security Controls for Computer Systems: Report of Defense Science Board Task Force on Computer Security," was published in 1970, and served as a primary input to the follow on effort to develop responsive DoD ADP security policies. This report was mentioned by Dr. Ware at the last seminar, and he has since taken the initiative in reprinting and making available copies of that report.

ADP security Task Force. A DoD Security Task Force was established under the Deputy Assistant Secretary of Defense (Security Policy) also in 1970. Its purpose was to identify, review and make necessary revisions to security policy directives in order to facilitate the utilization of advanced technology in automatic data processing systems. This charter, with the Defense Science Board report as input, shifted emphasis to the task of developing practical, realistic policy on a Department-wide basis, a significant undertaking especially at that time.

DoD Directive 5200.28 and DoD 5200.28-M. Following substantial effort by a number of participants, the basic policy documents were written, coordinated, and approved. DoD Directive 5200.28, entitled "Security Requirements for Automatic Data Processing (ADP) Systems," was published in December 1972 and its companion "ADP Security Manual" in January 1973 [2,3].

I shall briefly review some of the outlines of the documents to convey the security philosophy embodied therein.

The policies contained in these documents are designed to provide realistic, cost-effective parameters for the implementation of secure systems, *with specific recognition given to:* limitations in the technical state-of-the-art; operational considerations, particularly mission accomplishment; the wide variations within the universe of DoD and contractor computer systems; and, the overall potential cost impact of the requirements. Key illustrative provisions include the following:

--That classified material contained in an ADP system shall be safeguarded by the continuous employment of protective features in the system's hardware and software design and configuration, and by other appropriate administrative, physical, personnel, and communications security controls.

--That the basic ADP system reliability and integrity features must be augmented to assure that systems which process, store, or use classified data and produce classified information will, with reasonable dependability, prevent: a. Deliberate or inadvertent access to classified material by unauthorized persons, and b. Unauthorized manipulation of the computer and its peripheral devices;

--That the diversity and complexity of existing ADP systems as well as their demonstrated technical security weaknesses must be recognized and that alternative solutions to ADP system security problems are, in part, dependent upon the individual characteristics of the ADP system, and its usage;

--That the potential cost of the ADP system dictates that security policy be judiciously implemented, carefully managed, regularly reviewed, and continuously monitored to assure the most effective and economical use of the ADP system and related resources of the Department of Defense and of its contractors.

Toward those ends, the Directive provides for the application of administrative, physical, and personnel security measures to protect ADP systems, and includes the explicit assignment of responsibility for the testing, evaluation, and approval of such systems and for appointment of a responsible ADP System Security Officer for each ADP system approved for the processing of classified information.

POLICY CONTEXT

Authorities

Of course, our program is in implementation of and must be consistent
with requirements imposed by higher authorities. Congress has enacted a
number of significant statutes relating to our security program. Further-
more, the President, acting in his capacity as Chief Executive and as
Commander-In-Chief of the Armed Forces, has issued several Executive
Orders imposing security responsibilities upon the Secretary of Defense,
the most pertinent of which is E.O. 12065 (Figure 2) [4].

Particularly relevant to implementation of the order in the ADP environ-
ment is the information classification scheme; namely, that national
security information or material shall be classified in one of three
categories, Top Secret, Secret, or Confidential and no other categories
shall be used except as expressly provided by statute. Other designa-
tions coupled with one of these three categories pertain to access
restrictions only.

While the Executive Order focused primarily on the classification and
declassification of national security material and improving the balance
between the two competing principles of informing the public and preserv-
ing confidentiality, it also contains other pertinent, broad and generic
security policy requirements, most of which present problematic judgments
when applied to the ADP arena. For example, from Section 4:

- "No person may be given access to classified information unless
that person has been determined to be trustworthy and unless access is
necessary for the performance of official duties.

- All classified information and material shall be marked conspicu-
ously to put users on notice of its current classification status and,
if appropriate, to show any special distribution or reproduction restric-
tions authorized by this Order.

- Controls shall be established by each agency to ensure that
classified information is used, processed, stored, reproduced, and
transmitted only under conditions that will provide adequate protection
and prevent access by unauthorized persons."

Organizational Implementation

As these requirements are implemented by formal issuances down the
indicated organizational chains of command, they are elaborated upon and
generally specified as appropriate to more limited organizations and
environments. There are also built-in feedback mechanisms for the
evaluation of lower-level implementations. For example, in OSD, all DoD
Component implementations must be reviewed and certified as being consis-
tent with the basic DoD issuance. Similarly, the Executive Order provides
for an "Information Security Oversight Office" to assist the National
Security Council in monitoring implementation of the Order. One of its
functions is specifically to "oversee agency actions to ensure compliance
with this Order and implementing directives . . . . ."

The EO does not address computers per se. Our implementation, the Information Security Program Regulation, DoD 5200.1-R, /5/ doesn't either, except for paragraphs dealing with various media that may be associated with computer processing (e.g., punched cards, printouts, micro-forms). DoD Directive 5200.28 in essence represents our implementation of the EO insofar as the relatively unique problems posed by shared computer systems are concerned. The relationship between the two cannot be understated because much of the overall security guidance to be applied to the ADP environment is in 5200.1-R and is simply not duplicated in 5200.28. Therefore, in developing a system security plan, reference to both 5200.28 and 5200.1-R is required.

(Figure 3) Our ADP security program policies impact not only the DoD Components but also those ADP systems processing classified information among the 11,500 contractors in the Defense Industrial Security Program. As mentioned, this Program is administered by DoD on behalf of sixteen other Executive Branch Departments and Agencies, in addition to the DoD Components, and currently identified industrial general purpose ADP systems (about 700) represent a significant number of the total ADP systems subject to our ADP security policies.

"Other"/Special Access Programs (Figure 4)
So far the flow of implementation of policy is fairly straight forward. But there is always an "other," and as shown, there are basic.... four sets of Special Access Programs that impact the Informat.... .....rity Program:

NATO, where security procedures are based on International Treaty Requirements;

Requirements concerning access to and dissemination of Restricted Data and Critical Nuclear Weapon Design Information;

Special Access Programs for Foreign Intelligence under the cognizance of the Director of Central Intelligence or the National Communications Security Committee; and,

DoD "Special Access Programs" as such.

Our policy in this area is to utilize the standard classification categories to limit access to classified information on a "need-to-know" basis to personnel who have been determined to be trustworthy pursuant to the EO & NSC Directive, so that there will be no need to resort to formal Special Access Programs. That is, to avoid requiring the extraordinary procedures and controls, such as formal access determinations, special briefings, reporting procedures, and recorded formal access lists associated with Special Access Programs.

For simplicity's sake, consider these four as potential sources of additional security requirements in various areas which must be considered in system security planning, requirements that can range from the simple to the very complex and expensive.

KEY POLICY PROVISIONS

Background

I want to briefly describe some of the key policy provisions and struc-
ture for several reasons: first of all, some of the researchers have
been using as a point of departure, "old policy," that is, provisions
that have been superseded. Secondly, the current framework will be
relevant to any discussion of the fashion in which the output of the
Computer Security Initiative Program can be applied to an environment
where formal ADP security policies exist. Lastly, in covering key
provisions, I shall also indicate other aspects of our overall program
which relate to application of A-71.

Basic ADP Security Philosophy

In terms of security concepts, we do not view computer security as
fundamentally different from the protection of other information and
material. We do not orient on 100% security as feasible in this area --
even approaching that level is usually prohibitive in terms of cost or
constraint. Our approach is to relatively secure by employing security
barriers and measures in complementary combination (i.e., systematized
"defense in-depth") so that the cost/risk of penetration exceeds the
value or payoff of the penetration object, be it personal or classified
information, nuclear material or monetary assets. This "work factor"
approach involves identifying vulnerabilities (paths into the "system")
and erecting barriers generating a "work factor," in terms of cost/risk,
which exceeds the worth of the object(s) to be protected.

The end objective is an "acceptable level of risk determination" -- the
professional security judgment that the security subsystem generates
such a cost/risk work factor in a comprehensive, systematic and cost-
effective way. We feel the process through which this determination is
most effectively and validly made is the security analysis, test and
evaluation process (Figure 5), wherein both vulnerabilities and counter-
measures are systematically considered.

The computer security policy problem here is (Figure 6), there are no
generally accepted standards, criteria or even valid guidelines for
hardware/software security, yet this overall process is the basic tenet
of our policy. By contrast there are relatively clearcut guidelines and
minimum requirements in all the other security areas indicated. The end
result is that the process cannot now be executed with sufficient confi-
dence in terms of validity or reliability, let alone cost effectiveness.

It is precisely this problem to which I see the Computer Security Initia-
tive responding. Let me first, however, outline the policy framework
which I feel can effectively accommodate the Initiative Program concepts
as they are evolving -- as will be briefed during this seminar.

Policy Objective

As a point of departure here is the collective end objective (Figure 7).
The ADP system's collective security measures must, with reasonable
dependability, prevent both:

1.   access to classified material by unauthorized persons, __and__

2.   unauthorized manipulation of the ADP system.

Although we are protecting information, in this arena the ADP system as such must be protected. The "why" of it has been suggested -- currently available systems are penetrable and complex [e.g., 6,7,8]. Most significantly, penetration need not be executed at the time unauthorized access to classified information is effected. Rather, a penetration may be effected at one time and remain undetected for long periods of time prior to exploitation.

ADP System Security Modes (Figure 8)
In seeking to accommodate the hardware/software security problem with the need to operate, the need to employ ADP system to accomplish or support a multitude of defense missions, a set of alternatives evolved which may be viewed simply as alternative paths that involve the sorts of tradeoffs I mentioned at the beginning. Although not stated as such on the slides, one key variable, in the terms of this seminar, is the relative degree of "trustedness" insofar as the hardware/software security component is concerned. The modes involve basic tradeoffs between conventional security measures on one hand and hardware/software measures on the other. Viewed as alternatives along a continuum, as one moves from left to right relative hardware/software security responsibility increases, along with relatively increasing risk and uncertainty. In parallel, relative degree of security cost and constraint tends to decrease. The selection of one of these modes for a system is, of course, largely dependent upon the specific system, its functional requirements, its users and its environment, as to which mode is the most cost-beneficial.

As a further specification (Figure 9), let me relate these modes to the two policy requirements for access to classified material. Before an individual may be granted access to classified information: 1. he must have been granted a security clearance; and, 2. his access must be necessary for the performance of his official duties (i.e., he must have a "Need-to-Know"). In the manual world, both clearance and Need-to-Know determinations are normally made by humans in a fairly straightforward way. In the automated environment, however, this can vary. Moving again from left to right, clearance and Need-to-Know are determined prior to system access in the Dedicated Mode. In the System High Mode, clearance is determined before access, but Need-to-Know is not. The double line indicates a significant change in hardware/software security role -- to the right of the lines, it becomes one of preventing outright security violations and compromises. Now let's look at some specifics.

The Dedicated Mode, (Figure 10) at the far left, is the most clearly approvable type of system simply because the key security functions I noted are formed by comfortable, well understood conventional security measures. By definition, everyone with access to such a system has a

clearance and a Need-to-Know for everything then in the system. The
major protection burden is assumed by conventional personnel and physical
security measures and techniques which isolate the system from unautho-
rized personnel, pursuant to fairly clear policy requirements. Hardware/
software security role is minimized as a result.

The Full Multi-Level Security Mode (Figure 11), is at the other extreme.
There are some system users who have neither clearance nor Need-to-Know
for material contained in the system at the time of their access. In
this case, in direct contrast to the Dedicated Mode, both clearance and
Need-to-Know are determined by the ADP system. The separation of users,
their programs and files must be maintained by hardware/software security
mechanisms under operating system control, because it's all in the
computer and potentially accessible at the same time. In terms of
tradeoffs, the direct security costs and associated constraints on
system utilization are minimized (e.g., not all users with concurrent
access need be cleared to the highest levels; remote terminal areas need
not meet the physical security requirements of the central computer
facility; cpu (central processing unit) time and system availability are
not lost through sanitization procedures, and so on). But at the same
time, hardware/software security responsibilities are now maximized.
The major burden of key security functions falls upon hardware/software.

The "System High Mode" (Figure 12). The basic distinction between
Dedicated and System High is the matter of Need-to-Know. In both cases,
all users are cleared to the highest level. In the Dedicated Mode,
Need-to-Know is determined before actual system access is afforded to
users; in the System High Mode, it is determined by the ADP system
during access. It is established and maintained by hardware/software.
This mode is a more flexible, less constraining mode of operating an ADP
system than the Dedicated Mode. But, election of this mode requires the
development and implementation of hardware/software mechanisms to imple-
ment Need-to-Know.

The Controlled Mode (Figure 13) moves one step further along the continuum
and crosses that significant double line. Neither individual clearance
nor individual Need-to-Know are predetermined. But, in contrast to the
Multi-Level Security Mode, the important difference is a set of explicit
measures to reduce risk and vulnerability and to directly enhance or
even bypass hardware/software security measures under operating system
control.
Basically, the objective here is to provide a potentially approvable,
interim alternative to the more restrictive Dedicated and System High
Modes - a transition. But, one must take explicit steps, vice the
Multi-Level Mode, to reduce relative risk and vulnerability, and, pre-
ferably in combination, other steps to augment the system hardware/
software security posture. Examples of risk reduction are limits on the
range of clearance levels of users who have concurrent access (e.g.,
users of only two clearance levels). Actions that can concurrently
reduce relative vulnerability include restrictions on users capabili-
ties, such as providing query and response capability (Figure 14).

J-10

Application to Initiative Program Concepts

I think clearly the most important aspect of the foregoing is the rather clear potential linkage between modes, as a continuum of systems on the basis of relative required "trustedness," and efforts of this Computer Security Initiative Program, dealing with development of "trusted" ADP systems.

As this slide shows, (Figure 15) there is a clear correlation between the relative levels of protection that are evolving for purposes of evaluation and the continuum of modes. The left hand column will be treated in specifics by Grace Nibaldi and Peter Tasker $\sqrt{9}$,10 -- the general point I want to make here is that there is a clear potential relationship between the Initiative Program's efforts and the provisions of existing policy. Application of those efforts to real world ADP systems through existing policy is therefore neither remote nor obscure.

The notion here is that one might tentatively select a target system security mode on the basis of inherent security capabilities in a system during the initial stages of the risk assessment (Figure 16). There would follow detailed identification and assessment of a host of variables, both technical and non-technical peculiar to the individual system, any of which, in a tradeoff context, might change the relative security posture of the system "up" or "down" in the right-hand column; that is, with regard to the system security mode ultimately proposed for formal approval by the Designated Approving Authority. Jack Adams has developed a framework for enumerating critical security considerations that can be applied to the middle "interface" column $\sqrt{11}$.

The significance of this linkage is now limited to those systems processing classified information in DoD and in industry; as I'll suggest in a moment, that significance may be much more profound, depending upon the policy framework that ultimately evolves with regard to the computer security requirements of Transmittal Memorandum No. 1 to OMB Circular A-71.

From the policy interaction, let me turn briefly to the procedural -- how the expertise being developed within the Initiative Program might interface with the folks in the field who are currently tasked, and have been for some time, with evaluating and approving real world ADP systems.

As a point of departure, let me again refer to the general process that is a fundamental tenet of our policy (Figure 17). Recall that other than the "hardware/software" area indicated, criteria and requirements are relatively clear. Also given both resource limitations and the highly technical nature of the task, it appears most likely that formalized establishment of the Initiative Program's expertise will be at least initially centralized.

The technical expertise can be integrated into the test and evaluation process as shown here, by complementing the ongoing Component activities in the technical area. Recall that our policy explicitly delegates ADP system security approval authority to the DoD Components (and DLA for

contractor ADP systems). It is not our intention to change that -- the final approval must be on a system-by-system basis; that is, keyed to an individual system with its unique environment and functional requirements.

Though this is an old slide (Figure 18), it shows the place of technical advice, indicated in red, in the overall Component evaluation process. It also indicates our intent that the overall synthesis of the diverse parts of the analysis, together with the final decision to approve or not approve, lies with the appropriate Component Designated Approving Authority.

This overall notion might be kept in mind as I pursue the projection of classified arena concepts to a broader environment in ADP security.

A STRUCTURED CONCEPT FOR A-71 IMPLEMENTATION

Thus far, we have been discussing exclusively the policy framework for the protection of classified information in the ADP environment (Figure 19). As most of you are aware, the Office of Management and Budget promulgated much broader ADP security requirements in July 1978, specifically Transmittal *Memorandum No. 1* to OMB Circular A-71, entitled "Security of the Federal Automated Information Systems" /12/. This is a truly omnibus policy in that it is concerned with more than information security per se and more threats than just unauthorized disclosure of national security information. A-71 establishes a number of responsibilities and imposes a number of requirements on Executive Branch agencies.

A-71 Responsibilities and Requirements

To consider this document and its potential relationship to the Computer Security Initiative Program, let's first briefly review the scope and content of the program. First, it covers all Federal data and applications processed by computer.

This new program requires each Executive Branch Agency to:

- Assign responsibility for the security of each computer installation operated by or on behalf of the agency to a management official knowledgeable in data processing and security;

- Establish personnel security policies for all Federal and contractor personnel *involved* in the design, operation, or maintenance of, or having access to data in, Federal computer systems;

- Establish a management control process to assure that appropriate administrative, physical and technical safeguards are incorporated into all new computer applications and significant modifications to existing applications *(for applications deemed "sensitive," this includes: prior definition and approval of security specifications and the conduct, approval and certification of design reviews and application systems tests);

- Assure that appropriate security requirements are included in the specifications for the acquisition or operation of computer facilities or services;

- Conduct periodic risk analyses for each computer installation operated by or on behalf of the agency (at least every five years);

- Conduct independent periodic audits or evaluations and recertify the adequacy of the security safeguards of each operational sensitive application (at least every three years); and,

- Assure that appropriate contingency plans are developed and maintained to provide for continuity of operations should events occur

J-13

which prevent normal operations; periodically review and test these plans.

Also under the new program:

- The Department of Commerce will develop and issue computer system security standards and guidelines;

- The General Services Administration will issue policies and regulations for the physical security of computer rooms and assure that security requirements are included in agency procurements; and,

- The Civil Service Commission (now Office of Personnel Management) has established personnel security policies for Federal personnel associated with computer systems [13]. (Their guidelines also imply applicability to contractors.)

DoD Implementation Approach

The approach we are pursuing in Defense is one of essentially applying to the A-71 requirements the ADP security policy framework that has evolved in the classified arena over approximately the past decade. Essentially, (Figure 20) we envision first categorization of data and applications on the basis of criteria analogous to those that exist for classified national security information. Secondly, ADP systems are primarily categorized in terms of the data/applications processed, and then specific systems security requirements are directly derived primarily on a system basis. Incorporated, of course, is the multidisciplinary, systematic approach to implementation that characterizes the classified arena. A third essential ingredient, directly relevant to the computer security initiative program, is utilization of the currently authorized system security modes discussed above.

Let me quickly review the data and application categories that we have proposed in the intended sequence. Recall in this regard that Dr. Burrows (Director, Institute for Computer Sciences and Technology, NBS) earlier here called for development of a uniform structure for protection.

Sensitivity Categories -- Data & Applications (Figure 21)

ADP I, "Critical-Sensitive". DoD data and applications stored or processed in, or communicated, displayed or disseminated by, an Automatic Data Processing (ADP) System shall be categorized as ADP I when one or more of the following criteria are met:

- Top Secret National Security Information -- The data or applications require protection in the interest of national security, and the classification designation is "Top Secret" (DoD Regulation 5200.1-R);

- Mission Critical -- The data or applications are such that the denial of use, loss, compromise, disablement or unauthorized alteration thereof could reasonably be expected to directly and gravely degrade or

jeopardize the capabilities of a Military Department, the Joint Chiefs of Staff, a Defense Agency or a Unified or Specified Command to timely and effective discharge of their primary functions (DoD Directive 5100.1) in support of DoD emergency and/or war plans;

- Life Critical -- The data or applications are such that the denial of use, loss, compromise, disablement or unauthorized alteration thereof could reasonably be expected to directly and gravely jeopardize human life;

- Automated Decisionmaking Systems -- Applications, not otherwise included in the foregoing, which issue checks, requisition supplies or perform similar assets control functions, based on programmed criteria with little human intervention, wherein the potential loss or exploitable monetary value of the assets handled could exceed $10,000,000 per year.

ADP II, "Noncritical-Sensitive". DoD data and applications, which do not meet any of the foregoing criteria for category ADP I, shall be categorized as ADP II when one or more of the following criteria are met:

- Secret or Confidential National Security Information -- The data or applications require protection in the interest of national security, and the classification designation is either "Secret" or "Confidential" (DoD Regulation 5200.1-R);

- Mission Critical -- The data or applications are such that the denial of use, loss, compromise, disablement or unauthorized alteration thereof could reasonably be expected to degrade or jeopardize component command or major staff element capabilities to support timely and effective discharge of Military Department, OJCS, Defense Agency or U & S Command missions and functions;

- Privacy -- The data or applications involve personal information requiring protection pursuant to the Privacy Act of 1974 (DoD Directive 5400.7);

- FOIA Exemptions -- The data or applications (unclassified) have been determined to be exempt from public disclosure, consistent with the requirements of the Freedom of Information Act (FOIA) (Section VI, DoD Directive 5400.7);

- Automated Decisionmaking Systems -- Applications, not otherwise included in the foregoing, which issue checks, requisition supplies or perform similar assets control functions, based on programmed criteria with little human intervention, wherein the potential loss or exploitable monetary value of the assets handled could range between $1,000,000 and $10,000,000 per year.

ADP III, "Nonsensitive". All other DoD data and applications which do not meet the criteria for categories ADP I or ADP II as set forth above.

Sensitivity Categories -- ADP Systems (Figure 22)

ADP I, "Critical-Sensitive". ADP systems shall be categorized as ADP I
when either of the following criteria is met:

- ADP I Data or Applications -- The ADP system stores or processes
one or more sets of data or applications categorized as ADP I, "Critical-
Sensitive," pursuant to the criteria herein; or,

- Automated Decisionmaking Systems -- The ADP system handles "auto-
mated decisionmaking systems" wherein the aggregate total potential loss
or exploitable monetary value of assets handled collectively by the ADP
system's automated decisonmaking systems applications could exceed
$10,000,000 per year.

ADP II, "Noncritical-Sensitive". ADP systems, which do not meet any of
the foregoing criteria for category ADP I, shall be categorized as ADP
II when either of the following criteria is met:

- ADP II Data or Applications -- The ADP system stores or processes
one or more sets of data or applications categorized as ADP I; or,

- Automated Decisionmaking Systems -- The ADP system handles "auto-
mated decisionmaking systems" wherein the aggregate total potential loss
or exploitable monetary value of assets handled collectively by the ADP
system's automated decisionmaking systems applications could fall between
$1,000,000 and $10,000,000 per year.

ADP III, "Nonsensitive". All other ADP systems processing DoD data or
applications.

Sensitivity Categories -- Personnel Positions (Figure 23)

ADP I, "Critical-Sensitive". Positions of personnel requiring access to
ADP I DoD data or applications OR unescorted access to an ADP I ADP
system(s).

ADP II, "Noncritical-Sensitive". Positions of personnel requiring
access to ADP II DoD data or applications OR unescorted access to an ADP
II ADP system(s).

ADP III, "Nonsensitive". Positions of all other personnel requiring
access to DoD data or applications OR requiring unescorted access to an
ADP system containing DoD data or applications.

Now when we link the foregoing to the system security mode concepts
already presented, we have the capability to minimize personnel security
clearances for systems, based, in the terms of this seminar, on the
relative "trustedness" of the internal system security controls. For
example:

J-16

Adjustments for Position Sensitivity Categories (Figure 24)

1. "Multilevel and Controlled Mode" Systems -- The positions of
ADP System Users with access to systems already approved to operate in
either the "Controlled Security Mode" or the "Multilevel Security Mode"
pursuant to DoD Directive 5200.28 (or, for contractor ADP systems, DoD
Manual 5220.22-M) shall be designated in the position sensitivity cate-
gory commensurate with the most sensitive category of the DoD data or
application(s) they will access under system constraints.

2. "Temporarily Dedicated" Systems -- The positions of personnel
with access to ADP systems currently operating under procedures that
effect temporary dedication to different sensitivity categories at
different periods of time (also called "color changing" or "periods
processing") shall be designated in the sensitivity category commen-
surate with the most sensitive category of DoD data or application(s)
contained in the system during periods of each individual's access to
the system. In remotely accessed systems, this will include remote
terminal users wherein the remote terminal is disconnected during higher
sensitivity category processing periods.

3. "Output Only" -- The positions of ADP System User personnel
shall be designated in the position sensitivity category commensurate
with the category of only the system output they actually receive when:
(1) such personnel do not input to or otherwise directly interact with
the system (i.e., no "hands on" or other direct input or inquiry capa-
bility), and (2) the output products are either reviewed prior to
dissemination or otherwise determined to be properly identified as to
content, intended recipient and sensitivity category (i.e., systems
approved to implement this option pursuant to paragraph IV.C.5.b., DoD
Directive 5200.28 or for contractor ADP systems, paragraph 108, DoD
Manual 5220.22-M).

4. "Technical Review" -- The positions of personnel who design,
develop or generate DoD data or applications, or who generate input to
an ADP system containing DoD data or applications, shall be designated
in a less sensitive position category when (1) such personnel do not
have access to ADP systems containing higher sensitivity category data
or applications, and (2) when the product or input generated by such
personnel is subject to "Technical Review."

The most important consequence of the foregoing is that if we pursue
this concept then the need for "trusted" systems, just within Defense,
will expand from potentially 27% of our inventory (the subset that
processes classified information) of general purpose ADP systems to
100%. With Defense contractors, the requirement is expected to also
increase, although there is no basis for anticipating specific numbers.

Executive Branch Implementation

A-71 implementation from an Executive Branch-wide perspective generates a number of problems, particularly when data/application interchange among agencies and departments is considered, and most notably when contractors additionally are involved.

Personnel Security

The first and perhaps the simplest aspect of A-71 implementation, relatively speaking, that was promulgated was in the area of personnel security by the Office of Personnel Management (OPM), pursuant to OMB tasking [13]. As might be expected, the OPM criteria and guidelines are primarily keyed to the existing personnel management structure and are oriented on individual personnel positions. The foregoing concepts, however, are oriented on a system basis, and general personnel security requirements as well as other general requirements, may be derived from system security level and system security mode.

Specific requirements in the personnel security area are essentially open ended insofar as the scope of personnel security investigations, the standard which an individual must meet to be eligible for assignment to an ADP position and the adjudicative criteria by which the individual will be judged to determine whether the standard has been met. From an Executive Branch-wide perspective with regard to contractors, such a decentralized approach can result in an uncoordinated effort which (1) may not provide a uniform degree of fairness to the subjects of the investigative/adjudicative processes, (2) would not tend toward mutual and reciprocal acceptance of personnel security determinations among Federal agencies, and (3) cause confusion among firms performing on ADP contracts with more than one Federal agency (also possibly requiring duplicative or repetitive investigation of contractor employees to meet different scope and adjudication criteria).

With these very real and significant concerns in mind, we prepared correspondence for OMB, which the action office for A-71 in DoD has already formally dispatched. We specifically proposed that the majority of the problems cited could be avoided by following the single executive agency concept of the Industrial Security Program, established under the provisions of Executive Order 10865. The Executive Order recognized that conflicts and lack of uniformity would result if each government agency in the classified arena implemented its own industrial security program, and it therefore provided for the extension of the DoD program to include other departments and agencies. As a result, DoD has executive agreements with 16 other Executive Branch agencies to provide an industrial security program on a cost reimbursement basis. Consequently, standardized requirements and procedures have been issued and are uniformly implemented by participating agencies and contractor facilities [14]. Moreover, investigations are conducted in accordance with a standard investigative scope and are adjudicated centrally by the Defense Industrial Security Clearance Office under uniform adjudicative criteria. Records of clearances are also centrally maintained. The major benefit,

however, is that industry does not have to contend with 17 different government security programs.

We accordingly recommended that the implementation of the contractor employee personnel security requirements of A-71 be carried out by means of a modification of the Industrial Security Program.

Beyond Personnel Security

As I suggested, the personnel security aspect of A-71 is in many respects the simplest. The logical extension of the foregoing suggest additional possibilities.

For example, the Industrial Security Program does more than the indicated central personnel security clearance function; it also inspects specific contractor facilities and issues and records "contractor facility clearances"* on behalf of the 17 participating Federal Organizations. More to our concern here, for a decade our Industrial Security Representatives have been inspecting and approving contractor ADP systems that process classified information.

It takes little imagination, therefore, to suggest that the same logic that argues for serious consideration of centralized handling of contractor personnel security likewise, or even more so in light of the innate complexity of the total A-71 tasks, suggests equally serious consideration be given to at least uniform handling of contractor ADP systems and related protection of Federal government data and applications pursuant to A-71.

I would further suggest that if two of the notions outlined above were specifically included, the resultant practical framework for a total program implementation, both in and out of government, would be substantially simplified. That is, if: 1. a categorization scheme for data and applications were implemented government-wide and 2. if mode concepts were adopted, then a ready-made policy framework would be rather easily created. It would further provide for substantial effort being directed to the truly difficult issues in A-71, whereas by contrast, the absence of such a framework would generate questions and issues that could be the subject of virtually endless debate, to the detriment of meaningful progress on effectively implementing the other charges of A-71. And if there is one consistent problem throughout the ten-year history of our classified ADP security program, it is a shortage of manpower -- that is, manpower per se; never mind the issue of the qualification of the people.

---

*   "An administrative determination that, from a security viewpoint, a facility is eligble for access to classified information of a certain category (and all lower categories)" [14].

A measure of precedent for a categorization scheme for Federal data and applications exists in the OPM guidelines by virtue of their correlating varying personnel security requirements to those established by Executive Order 10450 for the traditional national security area. Such categorization would provide, unlike the privacy area, discrimination between relative sensitivity and relative allocation of security resources generically.

Addition of the standardized mode concepts, with the above, would substantially simplify the overall risk assessment process at the general level and permit focus on those complex security aspects which are truly installation and system dependent.

Further, at least within Defense and among Industrial Security Program contractors, such an approach would eliminate a substantial portion of learning curve costs for people working with those ADP systems by employing a framework that has been in being for almost a decade. It's not perfect, but it has had a long "debugging" period.

In a phrase, it thus would provide "one face to industry" comprehensively. It would concurrently provide an implementation framework for the handling of intra-government flow of data and applications processed by computer, among Executive Branch agencies and departments, a flow which I understand is not insignificant in volume.

CONCLUSION

In summary, there is a policy framework which has evolved in computer security over a ten-year period that is in effect within the Department of Defense and in the Defense Industrial Security Program. Moreover, on the industrial side, there is an in-being, nation-wide system that has likewise been in operation for about decade. Implementation along the lines suggested above would virtually solve a number of serious potential problems relating to Government interface with the private sector and the intra-government flow of data and applications processed by computer. It would also provide for direct application of the "trusted systems" being developed through the Computer Security Initiative Program.

The mapping of such a proven and rather well accepted policy framework to A-71, particularly commonly accepted and operationally defined notions of "cleared people" and "approved systems," I believe warrants serious consideration and further exploration for the reasons given--I would most appreciate your views on this.

REFERENCES

1. Security Controls for Computer Systems: Report of Defense Science Board Task Force on Computer Security, published by the Rand Corporation for the Office of the Director of Defense Research and Engineering (Rand Report #R-609), February 11, 1970.

2. "Security Requirements for Automatic Data Processing (ADP) Systems," Department of Defense Directive 5200.28, December 18, 1972, as amended (Change 2, April 29, 1978).

3. "ADP Security Manual: Techniques and Procedures for Implementing, Deactivating, Testing, and Evaluating Secure Resource-Sharing ADP Systems," Department of Defense Manual DoD 5200.28-M, January 1973, as amended (Change 1, June 25, 1979).

4. "National Security Information," Executive Order 12065, The Federal Register, July 3, 1978.

5. "Information Security Program Regulation," Department of Defense Regulation DoD 5200.1-R, December 1978.

6. Branstad, D.K., "Privacy and Protection in Operating Systems," Operating Systems Review, VII, 1 (January 1973).

7. Stryker, D.J., "Subversion of a 'Secure' Operating System," Naval Research Laboratory, Washington, D.C. 20375, NRL Memorandum Report 282 (June 1974).

8. Abbott, R.P., et al., "Security Analysis and Enhancements of Computer Operating Systems," National Bureau of Standards, Washington, D.C. 20234, Report NBSIR 76-1041 (April 1976).

9. Nibaldi, G.H., "Proposed Technical Evaluation Criteria for Trusted Computer Systems," Report #M79-225, The Mitre Corporation, Bedford, Mass., October 25, 1979.

10. Nibaldi, G.H., "Specification of a Trusted Computing Base (TCB)," Report #M79-228, The Mitre Corporation, Bedford, Mass., November 30, 1979.

11. Adams, J.A., "Computer Security Environmental Considerations," Federal Systems Division, International Business Machines Corporation, Arlington, Va. 22209, (Final Draft) August 15, 1979.

12. "Security of Federal Automated Information Systems," Transmittal Memorandum No. 1 to OMB Circular No. A-71, Office of Management and Budget, Executive Office of the President, Washington, D.C. 20503, July 27, 1978.

13. "Personnel Security Program for Positions Associated with Federal Computer Systems," FPM (Federal Personnel Manual) Letter 732-7, Office of Personnel Management, Washington, D.C. 20415, November 14, 1978 (Subsequently incorporated in the Federal Personnel Manual as Section 9, Subchapter 1, Chapter 732).

14. "Industrial Security Manual for Safeguarding Classified Information," Department of Defense Manual DoD 5220.22-M, October 3, 1977, as amended (Change 2, April 5, 1979).

COMPUTER SYSTEM
MANAGEMENT TRADEOFFS

COST

EFFICIENCY ←→ EFFECTIVENESS
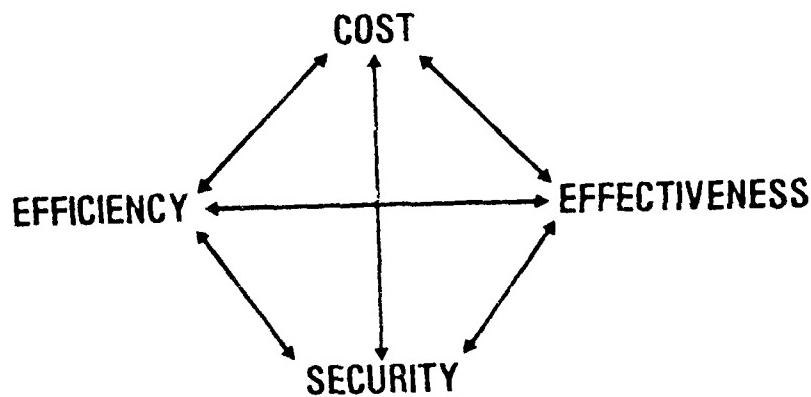
SECURITY

1

DEVELOPMENT AND IMPLEMENTATION OF ADP SECURITY POLICY
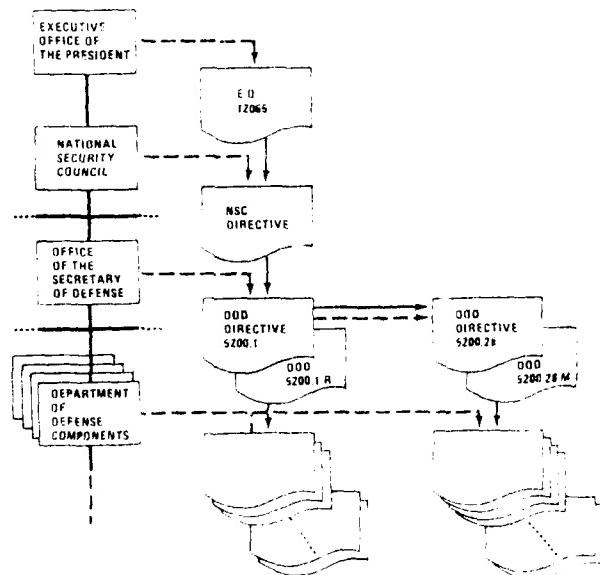
128

DEVELOPMENT AND IMPLEMENTATION OF ADP SECURITY POLICY
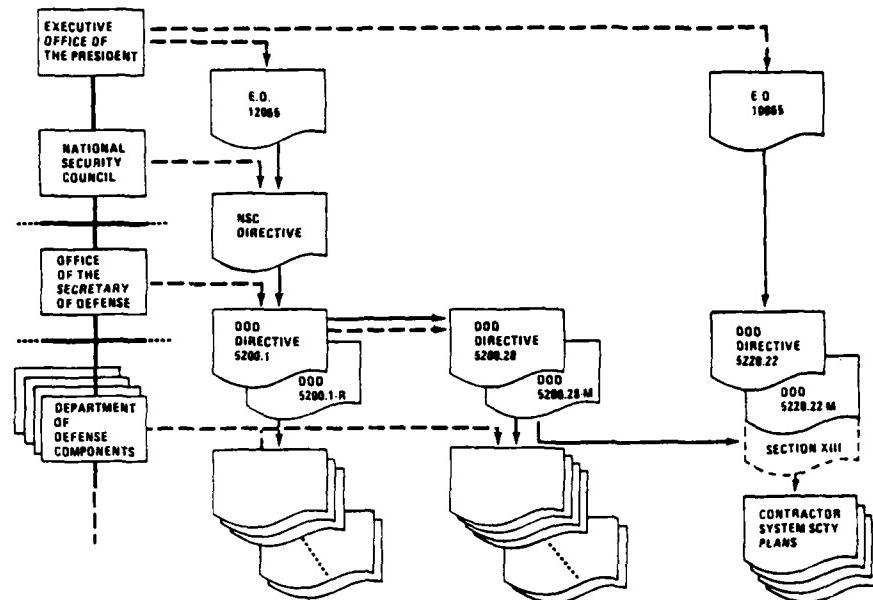
3

DEVELOPMENT AND IMPLEMENTATION OF ADP SECURITY POLICY
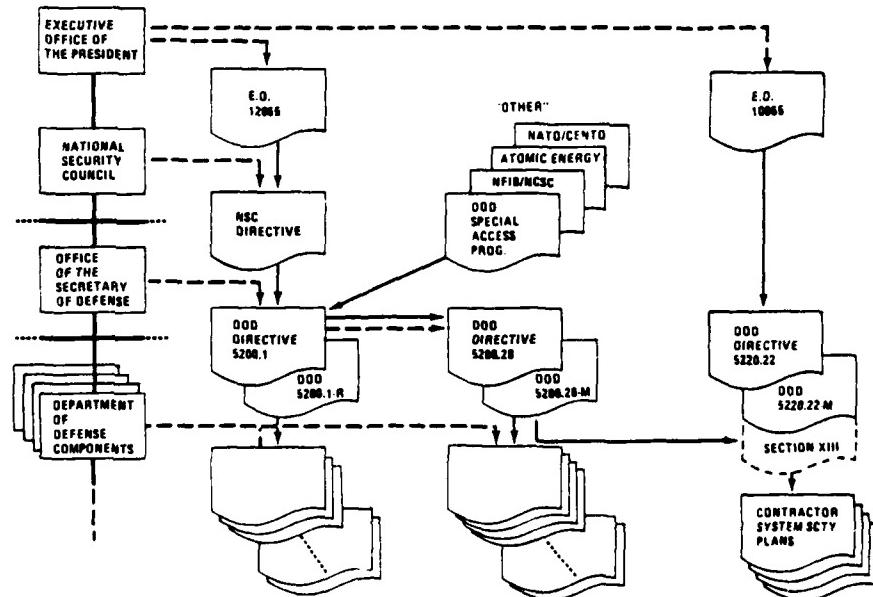
J-23

4

5

J-24
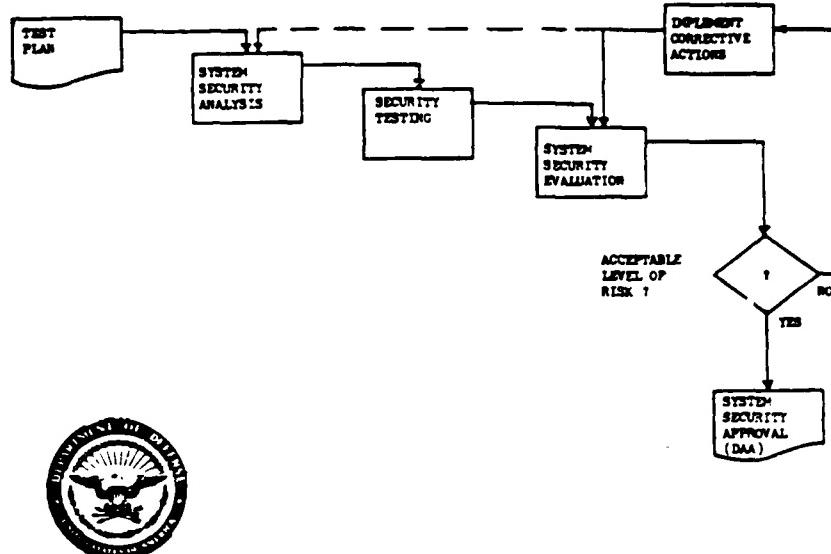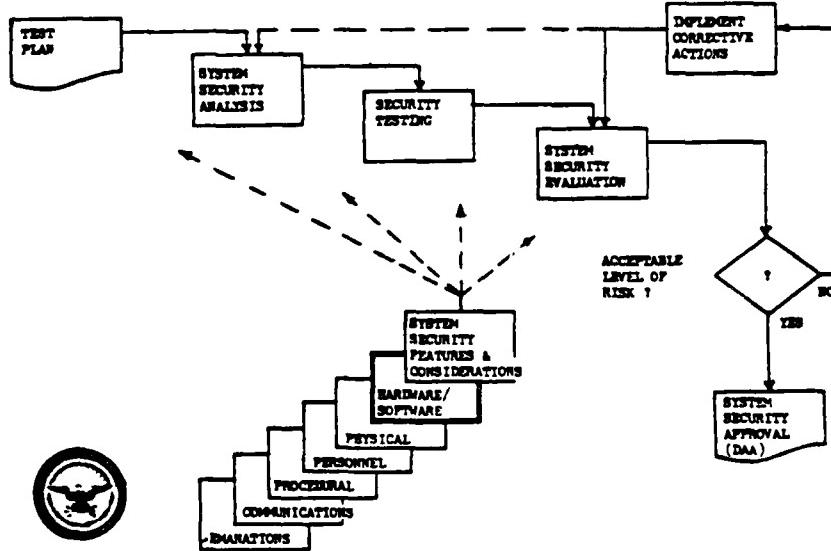
6

THE BASIC ADP SYSTEM RELIABILITY AND INTEGRITY FEATURES
MUST BE AUGMENTED TO ASSURE THAT SYSTEMS WHICH PROCESS,
STORE, OR USE CLASSIFIED DATA AND PRODUCE CLASSIFIED
INFORMATION WILL, WITH REASONABLE DEPENDABILITY, PREVENT:

A. DELIBERATE OR INADVERTENT ACCESS TO CLASSIFIED MATERIAL
    BY UNAUTHORIZED PERSONS, AND

B. UNAUTHORIZED MANIPULATION OF THE COMPUTER AND ITS
    ASSOCIATED PERIPHERAL DEVICES.

7

SPECTRUM OF ADP SYSTEM SECURITY MODES
REQUIREMENTS AND TRADEOFFS

ADP SYSTEM SECURITY MODE	"DEDICATED"	"SYSTEM HIGH"	"CONTROLLED"	"MULTI-LEVEL"

GENERIC TRADEOFFS.

INCREASING HARDWARE/SOFTWARE
SECURITY ROLE · INCREASING LEVEL
OF RISK AND UNCERTAINTY

DECREASING CONVENTIONAL
SECURITY COST/CONSTRAINT ON
ADP SYSTEM UTILIZATION.

SPECTRUM OF ADP SYSTEM SECURITY MODES
REQUIREMENTS AND TRADEOFFS

ADP SYSTEM SECURITY MODE	"DEDICATED"	"SYSTEM HIGH"	"CONTROLLED"	"MULTI-LEVEL"
HARDWARE/SOFTWARE SECURITY ROLE.	NIL	NEED-TO-KNOW	CLEARANCE (AUGMENTED) & NEED-TO-KNOW	CLEARANCE AND NEED-TO-KNOW

GENERIC TRADEOFFS.

INCREASING HARDWARE/SOFTWARE
SECURITY ROLE   INCREASING LEVEL
OF RISK AND UNCERTAINTY

DECREASING CONVENTIONAL
SECURITY COST/CONSTRAINT ON
ADP SYSTEM UTILIZATION.

9

SPECTRUM OF ADP SYSTEM SECURITY MODES
REQUIREMENTS AND TRADEOFFS

ADP SYSTEM SECURITY MODE	"DEDICATED"	"SYSTEM HIGH"	"CONTROLLED"	"MULTI-LEVEL"
GENERAL SECURITY REQUIREMENTS.				
PHYSICAL AND PERSONNEL.				
CENTRAL COMPUTER FACILITY	HIGH			
REMOTE TERMINAL AREAS	HIGH			
COMMUNICATION LINKS.	HIGH			
HARDWARE/SOFTWARE SECURITY ROLE:	NIL	NEED-TO-KNOW	CLEARANCE (AUGMENTED) & NEED-TO-KNOW	CLEARANCE AND NEED-TO-KNOW

GENERIC TRADEOFFS.

INCREASING HARDWARE/SOFTWARE
SECURITY ROLE   INCREASING LEVEL
OF RISK AND UNCERTAINTY

DECREASING CONVENTIONAL
SECURITY COST/CONSTRAINT ON
ADP SYSTEM UTILIZATION:

10

SPECTRUM OF ADP SYSTEM SECURITY MODES
REQUIREMENTS AND TRADEOFFS

ADP SYSTEM SECURITY MODE:	"DEDICATED"	"SYSTEM HIGH"	"CONTROLLED"	"MULTI-LEVEL"
GENERAL SECURITY REQUIREMENTS.				
PHYSICAL AND PERSONNEL.				
CENTRAL COMPUTER FACILITY	HIGH			HIGH
REMOTE TERMINAL AREAS	HIGH			VARIABLE
COMMUNICATION LINKS.	HIGH			VARIABLE
HARDWARE/SOFTWARE SECURITY ROLE:	NIL	NEED-TO-KNOW	CLEARANCE (AUGMENTED) & NEED-TO-KNOW	CLEARANCE AND NEED-TO-KNOW

GENERIC TRADEOFFS.

INCREASING HARDWARE/SOFTWARE
SECURITY ROLE - INCREASING LEVEL
OF RISK AND UNCERTAINTY

DECREASING CONVENTIONAL
SECURITY COST/CONSTRAINT ON
ADP SYSTEM UTILIZATION:

||

SPECTRUM OF ADP SYSTEM SECURITY MODES
REQUIREMENTS AND TRADEOFFS

ADP SYSTEM SECURITY MODE:	"DEDICATED"	"SYSTEM HIGH"	"CONTROLLED"	"MULTI-LEVEL"
GENERAL SECURITY REQUIREMENTS.				
PHYSICAL AND PERSONNEL.				
CENTRAL COMPUTER FACILITY	HIGH	HIGH		HIGH
REMOTE TERMINAL AREAS	HIGH	HIGH		VARIABLE
COMMUNICATION LINKS.	HIGH	HIGH		VARIABLE
HARDWARE/SOFTWARE SECURITY ROLE.	NIL	NEED-TO-KNOW	CLEARANCE (AUGMENTED) & NEED-TO-KNOW	CLEARANCE AND NEED-TO-KNOW

GENERIC TRADEOFFS

INCREASING HARDWARE/SOFTWARE
SECURITY ROLE - INCREASING LEVEL
OF RISK AND UNCERTAINTY

DECREASING CONVENTIONAL
SECURITY COST/CONSTRAINT ON
ADP SYSTEM UTILIZATION:

J-27

SPECTRUM OF ADP SYSTEM SECURITY MODES
REQUIREMENTS AND TRADEOFFS

ADP SYSTEM SECURITY MODE:	"DEDICATED"	"SYSTEM HIGH"	"CONTROLLED"	"MULTI-LEVEL"
GENERAL SECURITY REQUIREMENTS:				
PHYSICAL AND PERSONNEL:				
CENTRAL COMPUTER FACILITY	HIGH	HIGH	HIGH	HIGH
REMOTE TERMINAL AREAS	HIGH	HIGH	VARIABLE	VARIABLE
COMMUNICATION LINKS:	HIGH	HIGH	VARIABLE	VARIABLE
HARDWARE/SOFTWARE SECURITY ROLE:	NIL	NEED-TO-KNOW	CLEARANCE (AUGMENTED) & NEED-TO-KNOW	CLEARANCE AND NEED-TO-KNOW

GENERIC TRADEOFFS:

INCREASING HARDWARE/SOFTWARE SECURITY ROLE -- INCREASING LEVEL OF RISK AND UNCERTAINTY

DECREASING CONVENTIONAL SECURITY COST/CONSTRAINT ON ADP SYSTEM UTILIZATION:

RELATIVE TECHNICAL SECURITY POSTURE

Category:	Features:	Examples:

VULNERABILITY FACTORS (Technical) Example:

INITIAL TARGET MODE

Prog. Capabilities

1	DATA SECURITY	Most Current Sys
2	FUNCTIONAL SPECIFICATION	"New EXEC 8"
	REASONABLE PENETRATION RESULTS	"VS"
3	REASONABLE MODERN PROGRAMMING TECHNIQUES	MULTICS
	LIMITED SYSTEM INTEGRITY MEASURES	
4	FORMAL DESIGN SPECIFICATIONS	
	SYSTEM INTEGRITY MEASURES	
5	PROVEN DESIGN SPECIFICATIONS	KSOS
	VERIFIABLE IMPLEMENTATION	KVM
	LIMITED COVERT PATH PROVISIONS	
6	VERIFIED DESIGN	
	AUTOMATED TEST GENERATION	
	EXTENDED COVERT PATH PROVISIONS	
	REASONABLE DENIAL OF SERVICE PROVISIONS	

o DEDICATED
o SYSTEM HIGH
o CONTROLLED
o MULTILEVEL

15

RELATIVE TECHNICAL SECURITY POSTURE

Category:	Features:	Examples:

VULNERABILITY FACTORS (Technical) Example:

INITIAL TARGET MODE

Prog. Capabilities

1	DATA SECURITY	Most Current Sys
2	FUNCTIONAL SPECIFICATION	"New EXEC 8"
	REASONABLE PENETRATION RESULTS	"VS"
3	REASONABLE MODERN PROGRAMMING TECHNIQUES	MULTICS
	LIMITED SYSTEM INTEGRITY MEASURES	
4	FORMAL DESIGN SPECIFICATIONS	
	SYSTEM INTEGRITY MEASURES	
5	PROVEN DESIGN SPECIFICATIONS	KSOS
	VERIFIABLE IMPLEMENTATION	KVM
	LIMITED COVERT PATH PROVISIONS	
6	VERIFIED DESIGN	
	AUTOMATED TEST GENERATION	
	EXTENDED COVERT PATH PROVISIONS	
	REASONABLE DENIAL OF SERVICE PROVISIONS	

AF DATA SERVICE CENTER

o DEDICATED
o SYSTEM HIGH
o CONTROLLED
o MULTILEVEL

J-29

16

17

SYSTEM SECURITY ANALYSIS & APPROVAL PROCESS

J-30

18

NATIONAL LEVEL INTEREST

GAO REPORTS:

1976

- "IMPROVEMENTS NEEDED IN MANAGING AUTOMATED DECISIONMAKING BY COMPUTERS THROUGHOUT THE FEDERAL GOVERNMENT" (APR 76)
- "COMPUTER-RELATED CRIMES IN FEDERAL PROGRAMS" (APR 76)
- "MANAGERS NEED TO PROVIDE BETTER PROTECTION FOR FEDERAL AUTOMATIC DATA PROCESSING FACILITIES" (MAY 76)

SENATE COMMITTEE ON GOVERNMENT OPERATIONS:

- "COMPUTER ABUSES-PROBLEMS ASSOCIATED WITH COMPUTER TECHNOLOGY IN FEDERAL PROGRAMS & PRIVATE INDUSTRY" (JUN 76)

1977

- "COMPUTER SECURITY IN FEDERAL PROGRAMS" (FEB 77)

OMB:

- "SECURITY OF FEDERAL AUTOMATED INFORMATION SYSTEMS," TRANSMITTAL NO. 1 TO OMB CIRCULAR NO. A-71
  - DRAFT FOR COORDINATION (SEP 77)

1978
  - FINAL ISSUANCE (JUL 78)

PRESIDENT: INITIATIVE TO ATTACK FRAUD & WASTE

- DOD STEERING GROUP ON OVERSIGHT OF DEFENSE ACTIVITIES
  - SUBCOMMITTEE ON COMPUTER FRAUD

GAO REPORTS:

1979

- "AUTOMATED SYSTEMS SECURITY-FEDERAL AGENCIES SHOULD STRENGTHEN SAFEGUARDS OVER PERSONAL AND OTHER SENSITIVE DATA" (JAN 79)
- GAO LETTER TO SECDEF (MAR 79)

J-31

19

POLICY CONCEPT

- **CATEGORIZE: DATA/APPLICATIONS; SYSTEMS**

- **INCORPORATE MULTI-DISCIPLINARY, SYSTEMS APPROACH**

- **EMPLOY CURRENT SYSTEM SECURITY MODES**

20

ADP SYSTEMS

CAT I: — CAT I DATA/APPLICATION

CAT II: — CAT II DATA/APPL.

CAT III: — ALL OTHERS

22

DATA & APPLICATIONS

CAT I:
— TOP SECRET
— MISSION
— LIFE
— $10 M/YR.

CAT II:
— SECRET & CONF
— MISSION
— PRIVACY
— FOIA
— $1 - 10 M/YR.

CAT III:
— ALL OTHERS

21

ADJUSTMENTS

- TEMPORARY DEDICATION

- "MLS & CONTROLLED MODE"

- OUTPUT ONLY

- "TECHNICAL REVIEW"

- AGGREGATION

23

POSITIONS

CAT I — REQUIRED ACCESS TO:
CAT I DATA/APPL OR
SYSTEMS

CAT II — REQUIRED ACCESS TO:
CAT II DATA/APPL OR
SYSTEMS

CAT III — ALL OTHERS

J-33

COMPUTER SECURITY
AN INTEGRATED, MULTI-DISCIPLINARY APPROACH IS REQUIRED
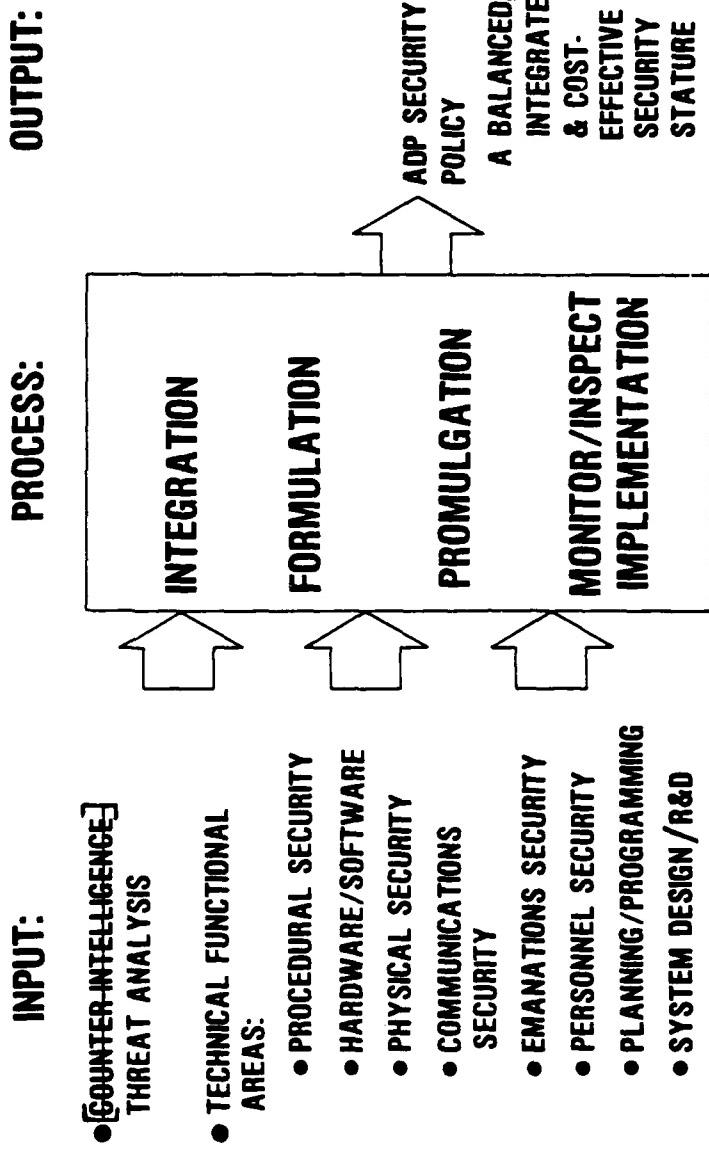
INPUT:

PROCESS:

OUTPUT:

• [COUNTER-INTELLIGENCE] THREAT ANALYSIS

• TECHNICAL FUNCTIONAL AREAS:

  • PROCEDURAL SECURITY
  • HARDWARE/SOFTWARE
  • PHYSICAL SECURITY
  • COMMUNICATIONS SECURITY
  • EMANATIONS SECURITY
  • PERSONNEL SECURITY
  • PLANNING/PROGRAMMING
  • SYSTEM DESIGN/R&D

INTEGRATION

FORMULATION

PROMULGATION

MONITOR/INSPECT
IMPLEMENTATION

ADP SECURITY POLICY

A BALANCED, INTEGRATED & COST-EFFECTIVE SECURITY STATURE

25

J-34

ADP-SECURITY REQUIREMENTS

FOR

E I F E L 2

LtCOL CERNY

CONTENTS

IT IS THE OBJECTIVE OF MY PRESENTATION, TO MAKE YOU FAMILIAR WITH
THE ADP-SECURITY REQUIREMENTS WE HAVE ESTABLISHED FOR OUR EIFEL 2
SYSTEM.

IN ORDER TO HELP YOU TO UNDERSTAND THESE REQUIREMENTS, THE FIRST
PART OF THIS PRESENTATION WILL BE A SHORT DESCRIPTION OF EIFEL 2
FROM AN OPERATIONAL AND TECHNICAL POINT OF VIEW AND THE SECOND
PART WILL BE USED TO GIVE YOU A CONCISE PRESENTATION OF THE SE-
CURITY REQUIREMENTS.

1. INTRODUCTION

THE GERMAN AIR FORCE CURRENTLY DEVELOPS A COMMAND, CONTROL AND
INFORMATION SYSTEM, WHICH WILL ASSIST THE MILITARY COMMANDERS
IN THE AIR FORCE AT ALL LEVELS OF COMMAND BY SUPPORTING THEIR
COMMAND AND CONTROL TASKS WITH MODERN INFORMATION PROCESSING
EQUIPMENT.

THIS SYSTEM WILL BE REALIZED IN SEVERAL STAGES, FIRST PARTS
ARE SCHEDULED TO BECOME OPERATIONAL IN 1984.

THIS ADP-SUPPORTED CCIS OF THE GERMAN AIR FORCE WILL CONSIST
OF A BASIC SYSTEM, CALLED EIFEL 2, WHICH ONE MIGHT CALL THE
ADP-WORKHORSE OF THE GAF-CCIS AND SEVERAL SOFTWARE-SUBSYSTEMS
WHICH WILL BE IMPLEMENTED ON EIFEL 2.

2. DESCRIPTION OF EIFEL 2

EIFEL 2 AS THE BASIC SYSTEM WILL PROVIDE FOR THE FOLLOWING
FUNCTIONS:

- COLLECTION, STORAGE, DISTRIBUTION AND DISPLAY OF STATUS IN-
  FORMATION
- REPORTING SYSTEM FOR ALL GAF-UNITS
- DISTRIBUTION OF ORDERS AND MESSAGES

- DECISION AIDS
- COMMON DATABASE FOR ALL SUBSYSTEMS
- PROCESSING CAPABILITY FOR ALL SUBSYSTEMS
- INTERFACE TO NATIONAL AND NATO SYSTEMS

AT THIS POINT IN TIME, WE ENVISION 3 SUBSYSTEMS:

- ONE SUBSYSTEM TO SUPPORT THE MISSION PLANNING PHASE FOR
  TACTICAL OFFENSIVE AIR-POWER BY PROVIDING SPECIAL APPLICATION
  FUNCTIONS.
  THIS SUBSYSTEM IS CALLED DISTEL;

- ONE SUBSYSTEM TO SUPPORT THE MISSION PLANNING PHASE FOR
  TACTICAL AIR-LIFT FORCES BY PROVIDING SPECIAL APPLICATION
  FUNCTIONS.
  THIS SUBSYSTEM IS CALLED SYLT;

- ONE SUBSYSTEM TO SUPPORT THE "MISSION PLANNING PHASE" FOR
  LOGISTIC FORCES BY PROVIDING SPECIAL APPLICATION FUNCTIONS.
  THIS SYSTEM IS CALLED SUSYLOG;

THIS PHILOSOPHY WITH ONE BASIC SYSTEM AND SEVERAL SUBSYSTEMS
WHICH WILL BE IMPLEMENTED ON THAT BASIC SYSTEM IN LINE WITH THE
OPERATIONAL REQUIREMENT FOR HIGH AVAILABILITY AND SURVIVABILITY
OF THE DATAPROCESSING POWER HAS RESULTED IN A SYSTEM ARCHITECTURE
WITH THE FOLLOWING MAIN CHARACTERISTICS:

- EIFEL 2 WILL CONSIST OF ADP-CENTERS, WHICH WILL BE DISTRIBUTED
  OVER THE TERRITORY OF THE FEDERAL REPUBLIC OF GERMANY;
- THERE WILL BE AN INTEGRATED DATABASE, WHICH WILL BE DISTRIBUTED
  OVER THESE ADP-CENTERS;
- ADP-CENTERS AND USERS WILL BE CONNECTED BY A PACKET-SWITCHED
  NETWORK;

A SCHEMATIC PRESENTATION OF THE SYSTEM ARCHITECTURE SHOWS
THAT EIFEL 2 WILL CONSIST OF

- HOST OPERATING CENTERS, TO PERFORM THE DATAPROCESSING TASKS,
- BASIC DATA PROCESSING CENTERS, WHICH HAVE IN GENERAL THE
  SAME DATAPROCESSING TASKS AS THE HOST OPERATING CENTERS, BUT
  ARE OF A SMALLER SIZE,
- TERMINALS, WHICH WILL BE PLACED DIRECTLY ON THE USERS DESK,
- A NETWORK CONNECTING USERS WITH DATAPROCESSING CENTERS AND
  USING PACKET-SWITCHING TECHNOLOGY,
- HOST INTERFACE PROCESSORS, TERMINAL INTERFACE PROCESSORS AND
  FOREIGN SYSTEMS INTERFACE PROCESSORS TO REALIZE THE DIFFERENT
  INTERFACES TO THE NETWORK,

LET ME GIVE YOU NOW A SHORT LOOK ON THE ESTIMATED NUMBERS OF
BASIC HARDWARE COMPONENTS WE THINK WE WILL NEED FOR EIFEL 2:

-    18 HOST OPERATING CENTERS
-    46 BASIC DATA PROCESSING CENTERS
-    35 PACKET-SWITCHES
-   240 TERMINAL INTERFACE PROCESSORS AND FOREIGN SYSTEMS
      INTERFACE PROCESSORS
- 2000 TERMINALS
- 1800 CRYPTO DEVICES
-    50 ALPHANUMERIC LARGE SCREEN DISPLAYS
-    50 GRAPHICAL LARGE SCREEN DISPLAYS

I WILL NOT GO INTO THE DETAILS OF THE SUBSYSTEMS, BECAUSE·AS
MENTIONED BEFORE, THE SUBSYSTEMS ARE ENVISIONED AS "APPLICATION
PACKAGES" WHICH W LL RUN ON EIFEL 2.

FROM THE SECUR'TY POINT OF VIEW, THEIR SECURITY REQUIREMENTS
WILL BE REALIZED THROUGH EIFEL 2, THEREFORE IN THE FOLLOWING

PART OF MY PRESENTATION, I WILL ONLY DISCUSS THE SECURITY RE-
QUIREMENTS FOR EIFEL 2.

3. ADP-SECURITY REQUIREMENTS

LET ME START WITH OUR DEFINITION OF SECURITY FOR EIFEL 2:

"SECURITY IS THAT CONDITION, WHICH GUARANTEES THE PROTECTION
OF CLASSIFIED INFORMATION FROM EITHER ACCIDENTIAL OR UNAUTHORIZED
INTENTIONAL DISCLOSURE, MODIFICATION OR DESTRUCTION AND EX-
CLUDES ANY INJURY TO THE SYSTEM BY ELEMENTS ENDANGERING ITS
SECURITY".

TO REACH THIS CONDITION, A WELL BALANCED SET OF SECURITY
MEASURES HAS TO BE DEVELOPPED IN THE AREAS OF
- PERSONNEL SECURITY
- PHYSICAL SECURITY
- ADMINISTRATIVE SECURITY
- COMMUNICATIONS SECURITY AND LAST BUT NOT LEAST
- ADP-SECURITY

FOR THE DISCUSSION TO FOLLOW, I WILL CONCENTRATE ON ADP-SECURITY
AND ONLY MENTION SHORTLY THE REQUIREMENTS FOR COMMUNICATIONS
SECURITY.

THE EIFEL 2 ADP-SECURITY REQUIREMENTS ARE BASED ON THE FOLLOWING
INPUTS:

- AN EVALUATION OF THE POTENTIAL THREATS TO THE SYSTEM
- AN EVALUATION OF THE OPERATIONAL REQUIREMENTS AND THEIR
  SECURITY IMPACT
- THE EXPERIENCE WITH OUR TESTSYSTEMS EIFEL 1 AND DISTEL 1
- AN EVALUATION OF THE EXISTING SECURITY AND PRIVACY REGULATIONS

THE FORMULATION OF THE SECURITY REQUIREMENTS HAS BEEN INFLUENCED

VERY MUCH BY THE WORK DONE BY OUR ADVISORY GROUP IABG AND
BY THE USAF/ESD ACTIVITIES(AS FAR AS PUBLISHED IN THE OPEN
LITERATURE).

A) THE THREAT

AS YOU CAN IMAGINE, MOST OF THE DATA WHICH WILL BE STORED,
PROCESSED AND DISTRIBUTED IN EIFEL 2 ARE HIGHLY SENSITIVE
AND VITAL FOR NATIONAL COMMAND AUTHORITIES AS WELL AS FOR
NATO. THEREFORE IT IS CERTAIN, THAT EIFEL 2, RESPECTIVE ITS
COMPONENTS, WILL BE A HIGH PRIORITY TARGET FOR ESPIONAGE
AND FOR SABOTAGE, BOTH IN PEACETIME AS WELL AS IN WAR AND
WILL BE SUBJECT TO ALL THE WELL-KNOWN THREATS OF ADP-SYSTEMS
LIKE

- THEFT
- FORGING OF DATA
- ERASURE OF DATA
- UNAUTHORIZED USE OF SYSTEM RESOURCES
- INTERCEPTION OF RADIATION FROM COMPUTERS, LINES AND
  TERMINALS
- TAPPING OF COMMUNICATION LINES
- CROSSTALK AND MISROUTING
- JAMMING

BUT COMPARED TO CONVENTIONAL ADP-SYSTEMS THE THREAT POTENTIAL
FOR EIFEL 2 IS GREATLY ENLARGED THROUGH TWO FACTORS:

- ITS ARCHITECTURAL CHARACTERISTICS LIKE

  + DECENTRALIZED PROCESSING WITH AN EVEN MORE DECENTRALIZED
    USER POPULATION OF APPROXIMATELY 4000 USERS AT 150
    DIFFERENT LOCATIONS IN THE FEDERAL REPUBLIC OF GERMANY
  + DISTRIBUTED DATABASE

+ PACKET-SWITCHED NETWORK

- OUR GEOPOLITICAL SITUATION, WHICH IS CHARACTERIZED BY

    + A CLOSE PROXIMITY TO THE WARSHAW PACT COUNTRIES

    + A SUPPOSEDLY GREAT NUMBER OF UNDERCOVER AGENTS, WELL
      TRAINED FOR ESPIONAGE AND SABOTAGE

THE IMPACT, THESE FACTORS HAVE ON THE RELIABLE AND SECURE
OPERATION OF EIFEL 2 JUSTIFIES IN OUR OPINION THE HIGH
PRIORITY THAT ADP-SECURITY HAS IN OUR WORK.

B) OPERATIONAL REQUIREMENTS

BEFORE TALKING ABOUT THE OPERATIONAL REQUIREMENTS AND THEIR
IMPACT ON ADP-SECURITY, IT HAS TO BE DETERMINED

- WHICH ARE THE MOST VALUABLE AND CRITICAL RESOURCES OF THE
  SYSTEM WHICH HAVE TO BE PROTECTED THROUGH ADP-SECURITY
  MEASURES AND

- WHAT IS THE REQUIRED GRANULARITY OF PROTECTION

IN A CCIS LIKE EIFEL 2 WHICH HAS TO SUPPORT MILITARY COM-
MANDERS IN ALL PHASES OF THE COMMAND AND CONTROL PROCESS,
THE DATA HAS THE HIGHEST VALUE FOR THE USER. THEREFORE IT
HAS TO BE PROTECTED

- AGAINST UNAUTHORIZED AND ACCIDENTAL OBSERVATION TO
  GUARANTEE ITS SECRECY AND

- AGAINST UNAUTHORIZED AND ACCIDENTIAL MODIFICATION AND
  DESTRUCTION TO RETAIN ITS INITIAL INTEGRITY OR SOUNDNESS
  THROUGHOUT THE OPERATION OF THE SYSTEM.

THERE IS NO DISCUSSION, THAT THE HARDWARE OF THE SYSTEM HAS

TO BE PROTECTED TOO, BUT THE NECESSARY PROTECTION MECHANISMS
ARE MAINLY A PART OF THE PROTECTION THROUGH PHYSICAL MEASURES
AND NOT PART OF ADP-SECURITY AS DISCUSSED IN THIS PRESENTATION.

BECAUSE EIFEL 2 WILL BE USED PRIMARILY IN AN INTERACTIVE
MODE WITH A DIALOGUE-LANGUAGE, THE GRANULARITY OF PROTECTION
SHOULD GO DOWN TO THE DATA-ITEM LEVEL.

NOW, IF WE LOOK AT THE LIST OF OPERATIONAL REQUIREMENTS FOR
EIFEL 2, THREE OF THEM HAVE A VERY STRONG IMPACT ON THE
ADP-SECURITY FUNCTIONS OF THE SYSTEM:

- AVAILABILITY
- TIMELINESS AND
- USER ACCEPTANCE

(1) AVAILABILITY

THE MAIN THRUST FOR THE SECURITY REQUIREMENTS COMES FROM
THE OPERATIONAL REQUIREMENT TO PROVIDE A SECURE AND CON-
TINUOUS SERVICE 24 HOURS A DAY/ 7 DAYS A WEEK. AS A CON-
SEQUENCE IT MUST BE POSSIBLE
- TO PROCESS DATA, PROGRAMS AND SUBSYSTEMS OF ALL SECURITY
  CLASSIFICATIONS/CATEGORIES SIMULTANEOUSLY
- TO SERVE USERS WITH DIFFERENT SECURITY CLEARANCES
  SIMULTANEOUSLY
- TO UPDATE DATA ITEMS WITH DIFFERENT SECURITY CLASSIFI-
  CATIONS/CATEGORIES SIMULTANEOUSLY IF A CHANGE IN THE
  REAL WORLD HAS OCCURED.

THIS REQUIRES A SYSTEM, WHICH ALLOWS THE SIMULTANEOUS
STORGAE AND PROCESSING OF DATA WITH DIFFERENT CLASSIFI-
CATIONS/CATEGORIES AND THE MANIPULATION OF THESE DATA
FROM REMOTE TERMINALS BY USERS HAVING DIFFERENT SECURITY

CLEARANCES.

AN OPERATING MODE LIKE " DEDICATED PROCESSING" IS NOT
ACCEPTABLE IN EIFEL 2 FOR OPERATIONAL REASONS.

(2) TIMELINESS

THE VALUE OF A CCIS DEPENDS LARGELY ON THE TIMELINESS
OF THE DATA IT CONTAINS. IN EIFEL 2 WE HAVE THE PHILO-
SOPHY OF THE "EVENT-ORIENTED" UPDATE. THIS MEANS, THAT
EACH TIME SOME STATUS IN THE REAL WORLD CHANGES, THERE
HAS TO BE AN IMMEDIATE UPDATE OF THE RESPECTIVE ENTRY
IN THE DATABASE.

BECAUSE SUCH A SITUATION CAN HAPPEN TO DIFFERENT DATA
OF DIFFERENT SECURITY CLASSIFICATION/CATEGORIES SIMUL-
TANEOUSLY, THERE AGAIN A MODE OF OPERATION LIKE "DEDI-
CATED PROCESSING" IS NOT ACCEPTABLE.

(3) USER ACCEPTANCE

ONE OF THE KEY FACTORS FOR THE ACCEPTANCE OF THE SYSTEM
BY THE USER IS THE "EASE OF USE".

EASE OF USE FROM A SECURITY POINT OF VIEW REQUIRES THAT
MECHANISMS WHICH ARE USED TO ENFORCE SECURITY MUST BE
DESIGNED IN SUCH A WAY, THAT THEY DO NOT OVERBURDEN THE
USER. OTHERWISE THERE WILL BE THE DANGER, THAT THE
SECURITY MECHANISMS WILL BE IGNORED OR CIRCUMVENTED.
(EXAMPLE: A TOO SOPHISTICATED PASSWORD PROCEDURE).

THE ORGANIZATIONAL INCONVENIENCIES FOR THE USER SHOULD
BE KEPT AT A MINIMUM. FOR EXAMPLE, IT SHOULD NOT BE
NECESSARY TO CLEAR USERS FOR OTHER CLASSIFICATIONS/
CATEGORIES THAN THEY NEED FOR THEIR WORK.

THEREFORE A MODE OF OPERATION LIKE "SYSTEM HIGH" IS NOT
ACCEPTABLE.

BY THE WAY, THERE ARE ADDITIONAL ARGUMENTS AGAINST A
"SYSTEM HIGH" MODE OF OPERATION.

FIRST, THE COST TO CLEAR MORE PEOPLE FOR HIGHER SECURITY
CLASSIFICATION THAN NECESSARY AND SECOND THE VIOLATION
OF THE PRINCIPLE OF "LEAST PRIVILEGE".

c) <u>EXPERIENCE WITH EIFEL 1 AND DISTEL 1</u>

WHEN WE STARTED WITH THE DEVELOPMENT OF BOTH SYSTEMS AS
TESTSYSTEMS IN THE LATE SIXTIES, ADP-SECURITY WAS NOT A
PRIMARY DESIGN GOAL, NEVERTHELESS THERE ARE SOME SECURITY
FEATURES - ACCORDING TO THE THEN AVAILABLE STATE OF THE
ART - WHICH PROOFED TO BE VERY EFFICIENT, ESPECIALLY IN THE
AREAS OF IDENTIFICATION, AUTHENTICATION AND AUTHORIZATION.
BUT THE MAIN EXPERIENCE WE GAINED WAS NOT IN THE AREA OF
ADP-SECURITY BUT IN THE AREAS OF PERSONNEL-, PHYSICAL-,
ORGANIZATIONAL- AND COMMUNICATIONS SECURITY.

d) <u>SECURITY AND PRIVACY REGULATIONS</u>

BASIS FOR MOST PART OF THE SECURITY REQUIREMENTS, WHICH WILL
BE DISCUSSED NOW, ARE THE EXISTING SECURITY REGULATIONS OF THE
GERMAN FORCES AND THE PRIVACY LAW OF THE FEDERAL REPUBLIC OF
GERMANY.

THE ADP-SECURITY REQUIREMENTS CAN BE ASSOCIATED TO TWO
CATEGORIES,

- GENERAL REQUIREMENTS AND
- SPECIAL REQUIREMENTS

FROM A USERS POINT OF VIEW, THERE ARE 5 GENERAL REQUIREMENTS,

WHICH WE THINK ARE ESSENTIAL WHEN TRYING TO PROTECT
CLASSIFIED/CATEGORIZED INFORMATION:

- A CLEAR DEFINITION OF THE LEVELS OF PROTECTION

  EIFEL 2 MUST BE CAPABLE TO PROTECT DATA-ITEMS,
  DATA-FILES WHICH HAVE DIFFERENT SECURITY CLASSIFICATIONS/
  CATEGORIES WHEREBY ONE RECORD CAN CONTAIN DATA-ITEMS OF
  DIFFERENT SECURITY CLASSIFICATIONS.
  THE SECURITY CLASSIFICATIONS RANGE FROM "RESTRICTED" TO
  "COSMIC TOP SECRET", EXAMPLES OF CATEGORIES TO BE HANDELD
  IN EIFEL 2 ARE "NATO" OR "CRYPTO". THE FULL SCOPE OF
  CATEGORIES IS NOT YET FINALLY DETERMINED.

- A DEFINITION OF THE ACCESS RULES

  THE BASIS FOR ACCESS PERMISSION IS THE ESTABLISHED NEED-TO-
  KNOW OF A USER OR HIS PRINCIPAL IN THE SYSTEM DERIVED FROM
  HIS OPERATIONA' TASKS.
  A USER, A PROGRAM OR A SYSTEM RESOURCE SHALL BE GRANTED
  ACCESS ONLY TO THAT CLASSIFIED/CATEGORIZED INFORMATION
  FOR WHICH HE HAS AN ESTABLISHED NEED-TO-KNOW AND THE
  APPROPRIATE ACCESS AUTHORIZATION. THE DEFAULT SITUATION
  SHALL BE LACK OF ACCESS.

- ADHERENCE TO THE PRINCIPLE OF LEAST PRIVILEGE

  IN THE ORGANIZATIONAL ENVIRONMENT OF EIFEL 2 AS WELL AS IN
  THE ADP-SYSTEM, THE PRINCIPLE OF LEAST PRIVILEGE MUST BE
  ADHERED TO. A USER, A PROGRAM OR A SYSTEM RESOURCE SHALL
  BE GRANTED ONLY THE SMALLEST POSSIBLE SET OF PRIVILEGES
  NECESSARY TO PERFORM ITS TASK.

- <u>INDIVIDUAL ACCOUNTABILITY</u>

  TO COMPLY WITH OUR SECURITY REGULATIONS AND TO ALLOW THE
  DETECTION OF BREACHES OF SECURITY, IT IS NECESSARY TO HAVE
  MECHANISMS TO TRACE THE ACTIONS OF THE USERS IN THE SYSTEM.
  THIS REQUIRES, THAT EVERY USER WHO IS WORKING WITH CLASSI-
  FIED/CATEGORIZED INFORMATION MUST BE MADE ACCOUNTABLE FOR
  HIS ACTIONS IN THE SYSTEM.

- <u>CONTINUITY OF OPERATION OF THE SECURITY MECHANISMS</u>

  THE COMPONENTS OF THE SYSTEM, WHICH FULFILL SECURITY
  FUNCTIONS MUST BE SWITCHED ON CONTINUOUSLY, THEY MUST BE
  CAPABLE OF SURVIVING ATTACKS AGAINST THEM. THE SECURE
  OPERATION OF THE SYSTEM AND ITS SECURITY MECHANISMS MUST
  BE DEMONSTRATED BY THE SYSTEM ITSELF BOTH AT REGULAR
  INTERVALS AND BY SPOT CHECKS.

NOW, LET ME DESCRIBE OUR SPECIAL SECURITY REQUIREMENTS:

- <u>CHANGE OF CLASSIFICATION/CATEGORY</u>

  SECURITY CLASSIFICATIONS AND CATEGORIES ARE ASSIGNED TO
  DATA-ITEMS, DATA-RECORDS AND DATA-FILES WHEN THEY ARE FIRST
  BROUGHT INTO THE SYSTEM. IT IS REQUIRED, THAT ONLY AUTHO-
  RIZED PERSONNEL CAN CHANGE OR DELETE THEM, THIS AT ANY
  TIME WITHOUT INTERRUPTING THE OPERATION OF THE SYSTEM.
  EVERY CHANGE OR DELETION MUST BE DOCUMENTED BY THE SYSTEM.

- <u>MARKING OF STORAGE- AND OUTPUT MEDIA</u>

  + STORAGE MEDIA MUST BE MARKED IN ACCORDANCE WITH THE
    SECURITY REGULATIONS, THE MARKING MUST BE ALSO READABLE
    BY THE SYSTEM;
  + LISTS, TABLES OR GRAPHIC DISPLAYS, WHICH ARE TO BE

K-13

PRINTED OUT BY THE SYSTEM, MUST BE MARKED AUTOMATICALLY
BY THE SYSTEM. ADDITIONALLY, THE SYSTEM MUST ATTACH AN
UNAMBIGUOUS REGISTRATION NUMBER TO EVERY PRINTOUT;

+ OUTPUTS ON DATA DISPLAY UNITS OR LARGE SCREEN DISPLAYS
MUST BE MARKED BY THE SYSTEM AUTOMATICALLY WITH THE
APPROPRIATE SECURITY CLASSIFICATION/CATEGORY.

- LOG-ON

PRIOR TO GET ACCESS TO THE SYSTEM AND TO THE CLASSIFIED/
CATEGORIZED INFORMATION, A USER MUST GO THROUGH THE
FOLLOWING STEPS:
+ IDENTIFICATION
+ AUTHENTICATION
+ AUTORIZATION

THERE ARE SOME ADDITIONAL REQUIREMENTS, WHICH CHARACTERIZE
THE LOG-ON FUNCTION:

+ THE NUMBER OF ATTEMPTS TO GET A LOG-ON FROM A TERMINAL
MUST BE LIMITABLE IN NUMBER AND TIME. ATTEMPTS EXCEEDING
THOSE PREDEFINED LIMITS MUST BE RECOGNIZED BY THE SYSTEM
AND BROUGHT FORWARD TO A SECURITY OFFICER FOR FOLLOW-ON
ACTIONS;

+ BEFORE A USER PUTS CLASSIFIED/CATEGORIZED INFORMATION
INTO THE SYSTEM BY MEANS OF HIS TERMINAL AND THROUGH THE
NETWORK, THE ADRESSED COMPUTER OR THE ADRESSED DATABASE
M UST IDENTIFY ITSELF TO HIM;

+ AFTER A SUCCESSFUL START OF A DIALOGUE, THE SYSTEM IS
REQUIRED TO CHECK THE USERS AUTHORIZATION FOR ACCESS IN
INTERVALS WHERE TIME AND SEQUENCE OF THE CHECKS ARE NOT
PREDETERMINABLE BY THE USER;

K-14

- ## STORAGE OF DATA AND PROGRAMS

THE SYSTEM MUST BE DESIGNED IN A WAY THAT IT HAS THE
DEMONSTRABLE CAPABILITY OF ENSURING THAT

+ INFORMATION CLASSIFIED AS "COSMIC TOP SECRET" IS STORED
  SEPARATELY FROM INFORMATION OF OTHER CLASSIFICATION;

+ NO USER OR PROGRAM MAY COME INTO A POSITION TO GAIN UN-
  AUTHORIZED INSIGHT OF OR ACCESS TO ANOTHER USERS
  CLASSIFIED INFORMATION AND

+ NO USER OR PROGRAM MAY COME INTO A POSITION TO MANIPU-
  LATE THE CLASSIFIED INFORMATION OR PROGRAM OF ANOTHER
  USER.

INFORMATION AND SOFTWARE, NECESSARY FOR THE IMPLEMENTATION
OF SECURITY MEASURES IN THE SYSTEM MUST BE STORED IN SUCH
A WAY THAT THEY CAN DEFINITELY NOT BE READ, CHANGED OR
ERASED BY OTHERS THAN THOSE RESPONSIBLE FOR SECURITY MATTERS.

- ## PROCESSING OF DATA

THE SYSTEM MUST ENSURE, THAT A NOT-PREDETERMINED MIX OF
DATA OF ALL SECURITY CLASSIFICATIONS/CATEGORIES CAN BE
PROCESSED SIMULTANEOUSLY. TO PREVENT SECURITY VIOLATIONS IT
MUST BE DEMONSTRATED THAT

+ THE SINGLE SECURITY PROPERTY AND

+ THE ✿ PROPERTY

CAN BE GUARANTEED BY THE SYSTEM.

IF EXTRACTS FROM DATA-FILES ARE REQUIRED AND SUCH EXTRACTS
MAY RECEIVE A LOWER SECURITY CLASSIFICATION THAN THE ORIGINAL
DATA-FILE, SPECIAL SECURITY MEASURES MUST BE TAKEN TO DOWN-
GRADE THE DATA IN A TRUSTABLE MANNER.

- ADMINISTRATION OF DATA

  THE SYSTEM MUST SUPPORT THE ADMINISTRATION OF CLASSIFIED/
  CATEGORIZED INFORMATION STORED, PROCESSED AND DISTRIBUTED
  IN THE SYSTEM BY LOGGING
  + ALL INPUTS
  + ALL STATES OF INTERNAL PROCESSING(ACCESSES, CHANGES ETC)
  + ALL OUTPUTS

- OUTPUT

  PRIOR TO PASSING CLASSIFIED/CATEGORIZED INFORMATION TO AN
  OUTPUT UNIT, THE SYSTEM MUST
  + POSITIVELY IDENTIFY THE USER AND THE OUTPUT UNIT IN-
    VOLVED
  + MAKE SURE THAT THE OUTPUT UNIT IS AUTHORIZED TO RECEIVE
    THAT OUTPUT.

  THE OUTPUT OF CLASSIFIED/CATEGORIZED INFORMATION TO UN-
  MANNED TERMINALS IS NOT PERMITTED.

- AUDITING AND SURVEILLANCE

  + ALL INFORMATION REQUIRED TO MONITOR SECURITY MUST BE DIS-
    PLAYED TO THE SECURITY OFFICER ON A SEPARATE TERMINAL;
  + THE SECURITY OFFICER MUST HAVE THE MECHANISMS TO
    - MONITOR THE ONGOING ACTIVITIES OF ALL TERMINALS
    - MONITOR THE ONGOING ACTIVITIES IN THE DATABASE
    - MONITOR THE HARDWARE;
  + THE SECURITY OFFICER MUST BE ABLE AT ANY TIME TO DENY
    USERS AND TERMINALS, EITHER SINGLY OR IN GROUPS, ACCESS
    TO THE SYSTEM AND TO THE DATA;
  + THE SYSTEM MUST HAVE THE MECHANISMS THAT USERS OR TER-
    MINALS, EITHER SINGLY OR IN GROUPS, CAN BE SWITCHED OFF

AT ANY TIME BY THE SECURITY OFFICER;

+ FOR AUDITING PURPOSES, THE FOLLOWING DATA SHOULD BE
  LOGGED:
  - EVERY ACCESS TO CLASSIFIED/CATEGORIZED INFORMATION,
  - EVERY MANIPULATION WITH SUCH INFORMATION,
  - EVERY LOG-ON OPERATION WHETHER SUCCESSFUL OR NOT,
  - EVERY LOG-OFF OPERATION,
  - EVERY CHANGE OF ACCESS PARAMETERS AND CLASSIFICATIONS/
    CATEGORIES

- PROTECTION OF TECHNICAL COMPONENTS

  + THE SYSTEM MUST ENSURE, THAT NO USER IS ABLE WITHOUT
    AUTORIZATION, TO DENY OR INTERRUPT THE SERVICES TO AN-
    OTHER USER;

  + THE SECURITY MECHANISMS OF THE SYSTEM SHOULD BE DESIGNED
    IN SUCH A WAY THAT THEY CANNOT BE BROKEN OR CIRCUMVENTED
    EVEN IF A PENETRATOR KNOWS HOW THEY WORK;

  + THE SYSTEM MUST AUTOMATICALLY BREAK CONNECTION WITH A
    TERMINAL IF
    - THE TERMINAL IS SWITCHED OFF,
    - THE POWER SUPPLY OF THE TERMINAL BREAKS DOWN,
    - THE COMMUNICATION LINE BREAKS DOWN,
    - WHEN, AFTER A SUCCESSFUL LOG-ON, THE TERMINAL RESTS
      UNUSED FOR A PREDETERMINED TIMEPERIOD,
    - THE CRYPTOGRAPHIC EQUIPMENT GOES OUT OF OPERATION,
    - THE NUMBER OF UNSUCCESSFUL LOG-ONS FROM A TERMINAL EX-
      CEEDS A PREDETERMINED LIMIT,

  + IF NECESSARY, A PROCEDURE FOR TERMINATION OF THE DIALOGUE
    MUST ERASE ANY CLASSIFIED/CATEGORIZED INFORMATION THAT
    MAY BE AVAILABLE IN THE USER TERMINAL,

  + ALL DEVICES AND COMMUNICATION LINKS MUST BE DESIGNED

K-17

IN A WAY THAT THEY

+ DO NOT EMIT INFORMATION OF A COMPREHENSIBLE NATURE

+ ARE INSENSITIVE TO JAMMING.

NOW LET ME FINISH THE LISTING OF ADP-SECURITY REQUIREMENTS
AND LET ME ADD SOME REQUIREMENTS FROM THE COMMUNICATION
SECURITY AREA.

- TRANSMISSION OF CLASSIFIED INFORMATION MUST BE EITER EN-
  CIPHERED OR - WHERE PERMISSABLE - THROUGH APPROVED CIRCUITS;
- THE FLOW OF INFORMATION OVER THE COMMUNICATION LINES MUST
  NOT ALLOW ANY INFERENCE ON THE TRUE NATURE OF THE TRAFFIC
  ON THE LINE;
- NO CLASSIFIED INFORMATION MAY APPEAR IN CLEAR LANGUAGE IN
  THE SWITCHING CENTERS;
- EVERYCOMPUTER IN THE SYSTEM MUST VERIFY THE AUTHORIZATION
  OF A USER ON ITS OWN INSTEAD OF RELYING ON THE VERIFICATION
  MADE BY ANOTHER COMPUTER.

4. CONCLUSION

I GAVE YOU A PRESENTATION OF THE ADP-SECURITY REQUIREMENTS FOR
EIFEL 2. I MUST ADMIT, THAT THIS IS A RATHER PRETENTIOUS CA-
TALOGUE OF REQUIREMENTS. WHEN WE DEVELOPED THIS CATALOGUE BEFORE
THE BEGINNING OF THE CONCEPTUAL PHASE OF EIFEL 2, OUR MAIN GOAL
WAS TO GIVE INDUSTRY A BASIS TO REALIZE MOST OF THE REQUIREMENTS
THROUGH TECHNICAL MEASURES IN ORDER TO KEEP MEASURES IN THE
AREAS OF PERSONNEL, INFRASTRUCTURE AND ORGANIZATION LOW.
NOW AT THE END OF THE CONCEPTUAL PHASE HOWEVER, WE FIND THAT
THERE ARE PROBLEMS IN THE TECHNICAL AREA WHICH WILL FORCE US -
AT LEAST IN THE FIRST STAGE OF EIFEL 2 - TO REALIZE SEVERAL OF
THOSE REQUIREMENTS THROUGH PERSONNEL, PHYSICAL AND ORGANIZATIONAL

MEASURES.

WE HOPE THAT A RISK-ASSESSMENT WE WILL START AT THE BEGINNING
OF THIS YEAR WILL LEAD US TO A WELL BALANCED SECURITY CONCEPT
FOR EIFEL 2 IN THE FIRST STAGE AND FOR THE STAGES TO FOLLOW.

THANK YOU.

LwFuDstKdo	German Air Force Information Systems Division	Stand:
AbtInfoVerarbLw		

EIFEL-2

ADP - SECURITY

REQUIREMENTS

EIFEL

GAF-CCIS PHILOSOPHY

UNCLASSIFIED

Stand:

LwFüDstKdo	E I F E L	UNCLASSIFIED
AbtInfoVerarbLw	FUNCTIONS	**Stand:**

O COLLECTION, STORAGE, DISTRIBUTION AND DISPLAY OF
   STATUS INFORMATION

O REPORTING SYSTEM FOR ALL GAF - UNITS

O DISTRIBUTION OF ORDERS AND MESSAGES

O DECISION AIDS

O COMMON DATABASE FOR ALL SUBSYSTEMS

O PROCESSING CAPABILITIES FOR ALL SUBSYSTEMS

O INTERFACE TO NATIONAL AND NATO SYSTEMS

LwFüDstKdo	EIFEL	UNCLASSIFIED
AbtInfoVerarbLw	GAF CCIS	**Stand:**

BASIC SYSTEM

   o EIFEL

   SUPPORTS THE COMMAND AND CONTROL PROCESS IN THE GAF BY
   PROVIDING BASIC DATAPROCESSING FUNCTIONS AND DATA IN
   ALL PHASES TO USERS AND SUBSYSTEMS

SUBSYSTEMS

   o DISTEL

   SUPPORTS THE MISSION PLANNING PHASE FOR OFFENSIVE
   AIR - FORCES BY PROVIDING SPECIAL APPLICATION FUNCTIONS

   o SYLT

   SUPPORTS THE MISSION PLANNING PHASE FOR TACTICAL AIR-LIFT
   FORCES BY PROVIDING SPECIAL APPLICATION FUNCTIONS

   o SUSYLOG

   SUPPORTS THE MISSION PLANNING PHASE FOR LOGISTIC FORCES
   BY PROVIDING SPECIAL APPLICATION FUNCTIONS

LwFuDstKdo		UNCLASSIFIED
AbtInfoVerarbLw	EIFEL ARCHITECTURAL CHARACTERISTICS	Stand:

DECENTRALIZED PROCESSING

DISTRIBUTED DATABASE

PACKET SWITCHED NETWORK

EIFEL

SYSTEM ARCHITECTURE
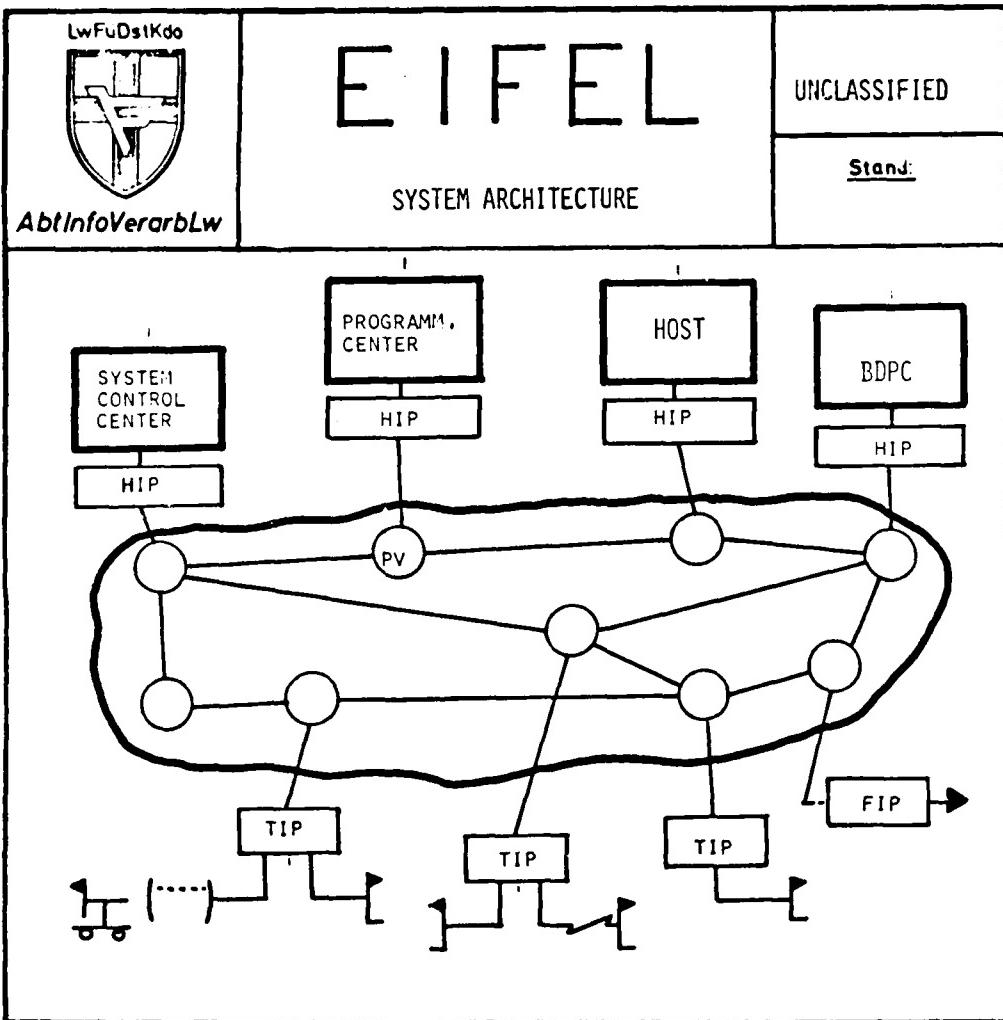
LwFuDstKdo

AbtInfoVerarbLw

UNCLASSIFIED

Stand:

LwFüDstKdo	EIFEL	UNCLASSIFIED
AbtInfoVerarbLw	SECURITY DEFINITION	<u>Stand:</u>

SECURITY IS THAT CONDITION WHICH

    O GUARANTEES THE PROTECTION OF CLASSIFIED INFORMATION

      IN EIFEL 2

    O EXCLUDES ANY INJURY TO EIFEL 2 BY ELEMENTS ENDANGERING

      ITS SECURITY

LwFuDstKdo	EIFEL	UNCLASSIFIED
AbtInfoVerarbLw	BASIS FOR ADP-SECURITY REQUIREMENTS	Stand:

O   POTENTIAL THREATS

O   OPERATIONAL REQUIREMENTS

O   EXPERIENCE EIFEL 1 / DISTEL 1

O   SECURITY / PRIVACY REGULATIONS

LwFuDstKdo	EIFEL	UNCLASSIFIED
AbtInfoVerarbLw	THREAT EXAMPLES	Stand:

O   THEFT

O   FORGING OF DATA

O   ERASURE OF DATA

O   UNAUTHORIZED USE OF SYSTEM RESOURCES

O   INTERCEPTION OF RADIATION

O   TAPPING OF COMMUNICATION LINES

O   CROSSTALK / MISROUTING

O   JAMMING

LwFüDstKdo	EIFEL	UNCLASSIFIED
AbtInfoVerarbLw	OPERATIONAL REQUIREMENTS	Stand:

O   AVAILABILITY

O   SURVIVABILITY

O   AUTONOMOUS MODE OF OPERATION

O   TIMELINESS

O   SECURITY

O   FLEXIBILITY

O   MOBILITY

O   USER ACCEPTANCE

LwFuDstKdo	EIFEL	UNCLASSIFIED
AbtInfoVerarbLw	SPECIAL REQUIREMENTS	Stand:

O CHANGE OF CLASSIFICATION/CATEGORY

O MARKING OF STORAGE- AND OUTPUT-MEDIA

O LOG - ON

O STORAGE OF DATA AND PROGRAMS

O PROCESSING OF DATA

O ADMINISTRATION OF DATA

O OUTPUT

O AUDITING AND SURVEILLANCE

O PROTECTION OF TECHNICAL COMPONENTS

LwFuDstKdo	EIFEL	UNCLASSIFIED
AbtInfoVerarbLw	GENERAL REQUIREMENTS	Stand:

O   LEVELS OF PROTECTION

O   CONTROLLED ACCESS

O   LEAST PRIVILEGE

O   INDIVIDUAL ACCOUNTABILITY

O   CONTINUITY OF OPERATIONS

Security Requirements, Design, and the Use of Trusted Software in a High Integrity Commercial Network

Dr. Thomas A. Berson
SYTEK, Inc.
Sunnyvale, CA

SYTEK EXPERIENCE WITH TRUSTED SYSTEMS

- MITRE Kernel I, II
- MULTICS, GUARDIAN
- SCOMP
- SATIN IV, SACDIN
- KSOS
- PSOS
- TACEXEC
- Other special programs

NETWORK CHARACTERIZATION

- Commercial, value added
- TDM, TDMA channels
- Packet switched

SECURITY MOTIVATION

1. Network self protection
2. Subscriber data protection
3. Government regulations (e.g. NSC-24)

SECURITY POLICY

"The network shall not misdeliver messages"

VULNERABILITIES & COUNTERMEASURES

- Design and implementation errors
  Trusted software—isolation and construction
- Active and passive channel tapping
  Link encryption
- Alteration of network environment
  Physical and personnel security
- Theft of network services
  Access control—authentication and authorization

L-2

TECHNIQUES FOR INCREASING CONFIDENCE
IN SOFTWARE

ABCDEF	Personnel integrity
CDEF	Configuration control
DEF	Formal statement of reliability criteria
BCDEF	Careful hierarchical design
DEF	Formal mathematical specification of design
CDEF	Peer and management inspection of design
EF	Proof that design conforms to desired criteria
BCDEF	Choice of appropriate programming language
BCDEF	Careful implementation of (proven) design
CDEF	Peer and management review of implementation
F	Proof that implementation conforms to (proven) design
BCDEF	Testing of system
CDEF	Controlled maintenance

CONCLUSION

1. Trusted software techniques can contribute to contemporary system design and construction.

2. More data is needed on the costs and benefits of trusted software techniques.

L-3

CURRENT STATUS OF COMPUTER SECURITY ACTIVITIES IN GERMANY

AND

RESULTS OF AN EVALUATION OF SPECIAL, KSOS, AND PSOS

Dr. Hans vor der Brück

Industrieanlagen- Betriebsgesellschaft mbH (IABG)

Ottobrunn b. Munich
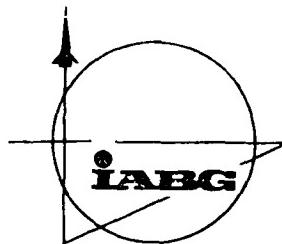
Current status of the security research
and development in the Federal Republic
of Germany

Requirements:

-     Requirements of the German Air Force
-     Requirements by the german privacy act
  - ✻   Secure Identification
  - ✻   Secure Authorisation
  - ✻   Unforgeable Access Control
  - ✻   Secure Protocol Functions
  - ✻   Secure Data Transmission

Current projects

Universities:

Karlsruhe: -   Investigations on abstract
                   models
              -    Implementation of a System
                   without formal Specifications,
                   but with Security as a main
                   design goal

Berlin: -   Formal Specifications of the
                   protocols of the ISO Reference
                   Model (Extension of Special)
              -    Two other groups are working
                   on Specification Languages

Stuttgart: -   EPOS: A Specification and Design
                   Technique for Real-time automation
                   systems

M-3

Current projects

SIEMENS

> Decision to build an operating system with security as one of its basic design issues.
>
> The concept phase has started
>
> Nothing is known about the underlying philosophy

IABG

- Since 1976 investigations in the area of computer security
- Lecturing problem awareness
- Penetrations of EDP-Systems
- Evaluation of security and audit software packages
- Monitoring status, progress and trends of computer security
- Technical advice in computer and communications security technology
- Development of requirements for the hardware and the software structure
- Proposals for the security concept of EIFEL II
- Investigations of Special and the Specifications of KSOS and PSOS

SPECIAL

Advantages:

- Formal specifications are an important step
  to provable secure systems

- A Special specification is relatively easy to convert
  into a program

- Special is relatively easy to learn

- Nonprocedural specification

- Strong typing

SPECIAL

Desirable Modifications and Improvements

- Clear definition of the relationship between the exceptions and effects

- Possibility to change V-functions only in the module in which they are defined

- Possibility of an exceptions-paragraph in hidden functions

- Expressions for the definition of sequencing and time conditions

- Hidden O- and OV-functions

Some results of the KSOS and PSOS
investigations

The aim of the investigations was:

To get some experience with Special specifications and the power of the tools.

Results:

a)   Formal mistakes, which had to be found by the
      described tools:

      -     PSOS

            * Hidden functions are referenced in other
               than their defining moduls

            * The Exceptions-part of V-, O- and OV-
               functions is frequently omitted

            * There are inconsistencies between the
               interface definitions and the references
               made from modules of a higher level
               to functions of a lower level

Formal mistakes, which had to be found by the described tools:

KSOS

*   There are inconsistencies between the EXTERNALREFS paragraph of the module KER and the modules in which the types and functions are defined.

*   The EXCEPTIONS part of V-, O- and OV-functions is often omitted

*   Hidden V-functions are referenced in other than the defining module

*   Variables of different types were equated or functions are called with parameters of the wrong type

Mistakes in the specifications, which couldn't be found
by the tools:

-       PSOS:

        *   In some functions the examination of the
                exception conditions is incomplete

        *   At creation time the user gets a wrong security
                level

        *   Loop indices are wrong at many places.
                At least one of these mistakes is critical
                to security

        *   Functions are called with wrong parameters

Mistakes in the specifications, which couldn´t be found by the tools:

- KSOS:

    * Inconsistencies between the verbal and the formal specifications

    * The function for the examination of the security rules is wrong

    * Sometimes functions are called with wrong parameters

    * Some important examinations of exception conditions are omitted

Our conclusions:

*   Formal specifications should be used for the
    development of secure operating systems

*   Special with some changes seems to be a proper
    tool

*   The tools for Special have to be improved

*   A formal specification with tools for the checking
    of the syntactical and some semantical properties
    does not guarantee a correct specification

*   Until more advanced and sophisticated tools become
    available, we have more confidence in systems based
    on a security kernel than in systems like PSOS

The Trusted Computing Base

P. S. TASKER
THE MITRE CORPORATION

Plan of Talk

TCBs and operating systems

Defining the protection policy

Developing TCB software

TCB function & structure

Evaluation Implications

How Does a TCB Relate to an Operating System?

Operating system basis

Supervisory and control services

Extended machine

Protection environment

Policy
Mechanism

Reference Monitor Concept

```
┌─────────┐        ┌──────────────┐        ┌─────────┐
│ SUBJECT │  ──→   │  REFERENCE   │   ──→  │ OBJECT  │
└─────────┘        │   MONITOR    │        └─────────┘
                   └──────────────┘
                          │  ↑
                          ↓  │
                   ┌──────────────┐
                   │ ACCESS RULES │
                   └──────────────┘
```
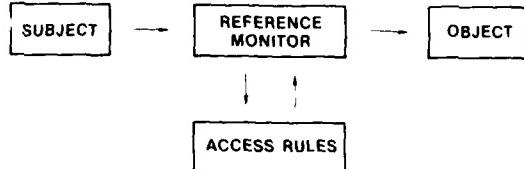
GOALS:
  COMPLETENESS
  ISOLATION
  VERIFIABILITY

Computer System Organization

```
         PROCESS BOUNDARY
            ¦                                        HUMAN
                                           ──        INTERFACE
   EDITOR   COMPILER   USER        OTHER
                       APPLICATION UTILITIES
                                                     SVC
                                           ──        INTERFACE

            OPERATING SYSTEM

                                           ──        HARDWARE
               HARDWARE                              INTERFACE
```
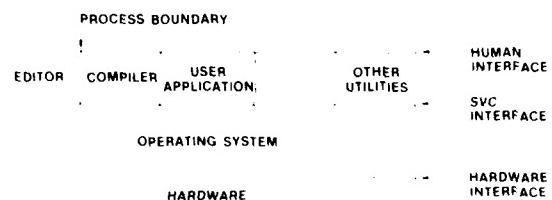
PER-PROCESS VIRTUALIZATION
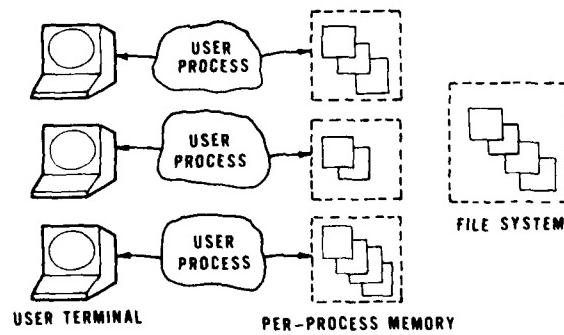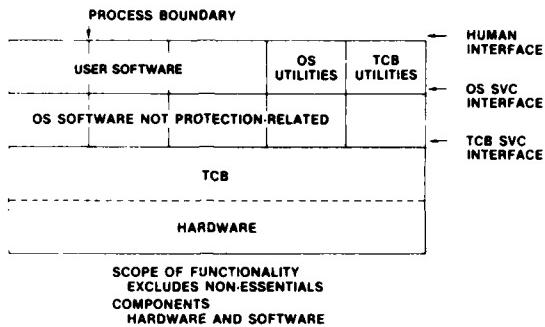
USER TERMINAL   PER-PROCESS MEMORY   FILE SYSTEM

Computer System Organization in Terms of a TCB

```
            PROCESS BOUNDARY
                                                           HUMAN
                                   OS      TCB      ◄──    INTERFACE
     USER SOFTWARE              UTILITIES UTILITIES
                                                           OS SVC
     OS SOFTWARE NOT PROTECTION-RELATED                ◄── INTERFACE

                                                           TCB SVC
                                                       ◄── INTERFACE
                        TCB

                     HARDWARE
```

SCOPE OF FUNCTIONALITY
  EXCLUDES NON-ESSENTIALS
COMPONENTS
  HARDWARE AND SOFTWARE

Hardware Support

Memory protection

Virtual memory

Tagged memory

Process control

I/O

Privileged states/domains

Three-Domain System
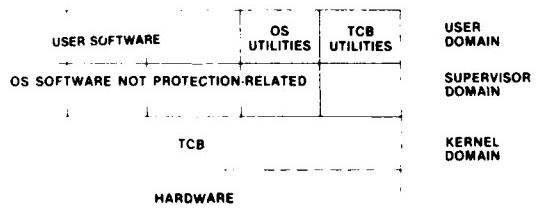
```
                                   OS      TCB         USER
     USER SOFTWARE              UTILITIES UTILITIES    DOMAIN

     OS SOFTWARE NOT PROTECTION-RELATED               SUPERVISOR
                                                       DOMAIN

                        TCB                            KERNEL
                                                       DOMAIN

                     HARDWARE
```

Plan of Talk

TCBs and operating systems

Defining the protection policy
  Distinguishing features
  Models

TCB function & structure

Evaluation implications

Protection Policy Elements

Subjects
  User
  Process

Objects
  Physical/logical disk
  Directory
  File
  Segment
  Page
  Record/field

Access rules
  Denial of service
  Security
  Integrity

Security Policy

Protection against unauthorized disclosure

Second-order concern:
  Confinement channels
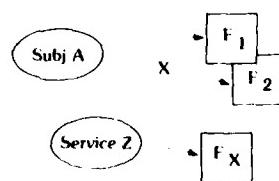
Confinement Problems

Untrusted program invoked to serve you (e.g., editor)

Direct

0. Collect data for its owner (if it has memory)
1. Copy information into file in owner's directory
2. Create temporary file granting owner access
3. Send message to owner

Indirect ("Covert")

4. Encode information in owner's copy of bill to customer
5. Use interlocks on files shared with owner to pass encoded information
6. Broadcast encoded information by varying
   A. Shared system control variables
   B. Shared resources (CPU use, paging)

Integrity Policy

Protection against unauthorized modification

Directly related to security

N-5

Policy Enforcement Strategy

Focus on security policy

Access control — protects containers

Flow control — protects contents of the containers

Access Control Model — Access Matrix

Example of a Set Theoretical Flow

Example of a Linear X Set Lattice — Military Policy

TS

S

C

U

Flow Control Model — Security Model

Subjects and objects are assigned security levels — SL( )

Axioms:

Simple security condition:	A subject can read an object iff SL(subject) . SL(object)
* - Property:	A subject can write an object iff SL(object) . SL(subject)
Activity:	Only objects that are active can be accessed.
Tranquility:	The SL of active objects cannot change.
Erasure:	When an object becomes inactive its contents must be erased.

Model Implications: Trojan Horse
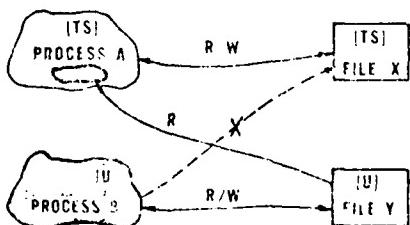
Plan of Talk

TCBs and Operating Systems

Defining the protection policy

TCB function & structure

Evaluation implications

TCB Functions

Establish a secure state

Control state transitions

Bind secure system to external environment

Establish a Secure State

Program Loaders

Initialization

Control State Transitions

Processes
  Create/delete
  Swap
  Send/receive IPC message
  Get/set status

Storage
  Create/delete
  Grant/revoke ownership/access
  Read/write data
  Get/set status

I/O
  Create/delete
  Grant/revoke ownership/access
  Read/write device
  Get/set status

Bind Secure System to External Environment

Operations interface

Device configuration control

Extension of policy

Facility dependent services

User space definition

Secure user interface

Preservation of secure state across discontinuities

Properties Of Trusted Software

May enforce a protection policy

May deliberately, selectively override policy
implemented by other trusted software

May perform functions that could indirectly
enable a policy to be violated

Kernelized System
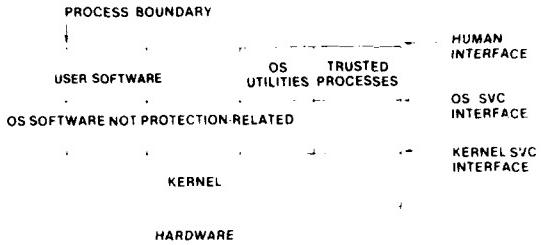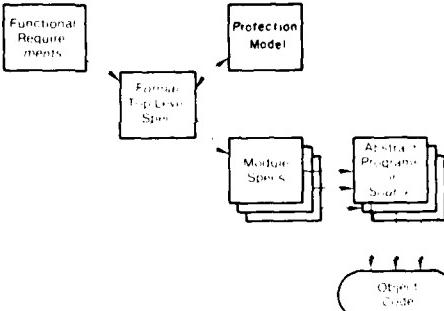
PROCESS BOUNDARY

USER SOFTWARE

OS UTILITIES   TRUSTED PROCESSES

HUMAN INTERFACE

OS SOFTWARE NOT PROTECTION-RELATED

OS SVC INTERFACE

KERNEL SVC INTERFACE

KERNEL

HARDWARE

Formal Specification in Software Development

Functional Requirements

Protection Model

Formal Top Level Spec

Module Specs

Abstract Programs or Spec

Object Code

Summary

TCBs and operating systems

  Provides a verifiable protection base for an operating system

Defining a protection policy

  Acts according to a precisely stated protection policy

TCB functions & structure

  Carries out certain relevant operating system functions
  Designed in methodical steps
  May be organized as a kernel and other trusted software

Evaluation Implications

TCB Specification:

Generic mechanism $\left\{ \begin{array}{l} \text{description} \\ \text{requirements} \end{array} \right.$

Evaluation Criteria:

Policy options

Mechanism features

Construction environment (assurance)

Levels of TCB Protection

0. No protection

1. Limited controlled sharing

2. Extensive mandatory sharing

3. Structured protection mechanism

4. Design correspondence

5. Implementation correspondence

6. Object code analysis

Documents Provide Focus
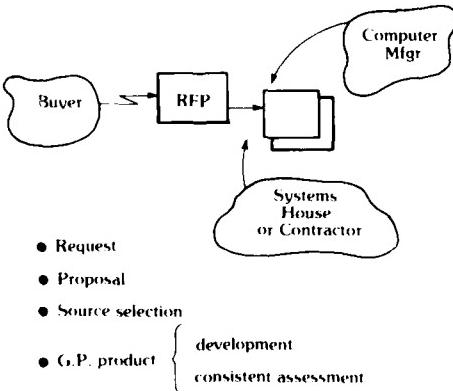
● Request

● Proposal

● Source selection

● G.P. product $\left\{ \begin{array}{l} \text{development} \\ \text{consistent assessment} \end{array} \right.$

The Trusted Computing Base

G. H. Nibaldi

The MITRE
Corporation

Reference Monitor Concept

SUBJECT — REFERENCE MONITOR — OBJECT

ACCESS RULES
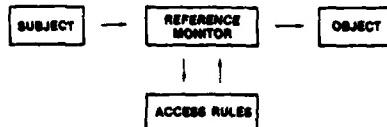
GOALS:
COMPLETENESS
ISOLATION
VERIFIABILITY

Plan of Talk

TCBs and operating systems

Defining the protection policy

Developing the TCB software

Computer System Organization
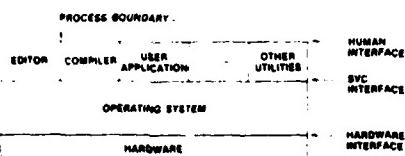
PROCESS BOUNDARY

EDITOR COMPILER USER APPLICATION OTHER UTILITIES   HUMAN INTERFACE

SVC INTERFACE

OPERATING SYSTEM

HARDWARE   HARDWARE INTERFACE

How Does a TCB Relate to an Operating System?

Operating system basis

Supervisory and control services

Extended machine

Protection environment

Policy

Mechanism

Computer System Organization in Terms of a TCB
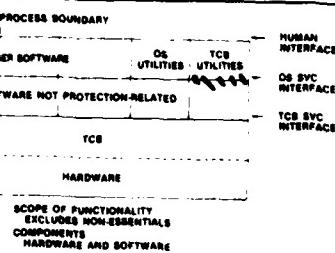
PROCESS BOUNDARY

USER SOFTWARE   OS UTILITIES   TCB UTILITIES   HUMAN INTERFACE

OS SVC INTERFACE

OS SOFTWARE NOT PROTECTION-RELATED

TCB SVC INTERFACE

TCB

HARDWARE

SCOPE OF FUNCTIONALITY
EXCLUDES NON-ESSENTIALS
COMPONENTS
HARDWARE AND SOFTWARE

Hardware Support

Memory protection

Virtual memory

Tagged memory

Process control

I/O

Privileged states/domains

Protection Policy Elements

Subjects
  User
  Process

Objects
  Physical/logical disk
  Directory
  File
  Segment
  Page
  Record/field

Access rules
  Denial of service
  Security
  Integrity

Three-Domain System

		OS UTILITIES	TCB UTILITIES	USER DOMAIN
USER SOFTWARE				
OS SOFTWARE NOT PROTECTION-RELATED				SUPERVISOR DOMAIN
		TCB		KERNEL DOMAIN
		HARDWARE		
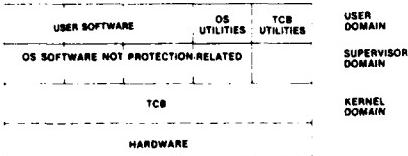
Denial of Service Policy

Protection against unauthorized disruption of service due to:
  Delay
  Crash
  Consumption of resources
  Destruction of information
  Masquerading information
  Masquerading services

Plan of Talk

TCBs and operating systems
Defining the protection policy
  Distinguishing features
  Models
Developing the TCB software

Security Policy

Protection against unauthorized disclosure

Second-order concern:
  Confinement channels

Confinement

Direct channels

Covert channels
 Storage
 Timing

Access Control Model — Access Matrix

OBJECTS

	O_1	O_2	O_3	...	O_N
S_1	—	X	X		—
S_2	X	X	—		—
S_3	X	X	X		—
.					
.					
.					
S_M	—	X	—		—

SUBJECTS

Integrity Policy

Protection against unauthorized modification

Directly related to security

Flow Control Model — Lattice Model

Subjects and objects are assigned to
security classes : Ai

A4
↑
A3
↑
Direction of flow ↑ A2
↑
A1

Policy Enforcement Strategy

Access control — protects containers

Flow control — protects contents of the containers

Example of a Set Theoretical Flow

{x,y,z}

{x,y}   {x,z}   {y,z}

{x}   {y}   {z}

Example of a Linear X Set Lattice — Military Policy

TS
↑
S
↑
C
↑
U

TCB Functions

Establish a secure state

Control state transitions

Bind secure system to external environment

Flow Control Model — Security Model

Subjects and objects are assigned security levels — SL( )

Axioms:

Simple security condition: A subject can read an object iff
SL(subject) ≥ SL(object)

• - Property: A subject can write an object iff
SL(object) ≥ SL(subject)

Activity: Only objects that are active
can be accessed.

Tranquility: The SL of active objects cannot change.

Erasure: When an object becomes inactive
its contents must be erased.

Establish a Secure State

Program Loaders

Initialization

Plan of Talk

TCBs and Operating Systems

Defining the protection policy

Developing the TCB software
  Functions
  Design Methodology

Control State Transitions

Processes
  Create/delete
  Swap
  Send/receive IPC message
  Get/set status
Storage
  Create delete
  Grant revoke ownership/access
  Read write data
  Get set status
I O
  Create delete
  Grant revoke ownership access
  Read write device
  Get set status

Bind Secure System to External Environment

Operations interface

Device configuration control

Expansion of policy

Facility dependent services

User space definition

Secure user interface

Preservation of secure state across discontinuities

Kernelized System

PROCESS BOUNDARY

PROCESS BOUNDARY

USER SOFTWARE	OS UTILITIES	TRUSTED PROCESSES	HUMAN INTERFACE
OS SOFTWARE NOT PROTECTION-RELATED			OS SVC INTERFACE
KERNEL			KERNEL SVC INTERFACE
HARDWARE			

Properties Of Trusted Software

May enforce a protection policy

May deliberately, selectively violate policy implemented by other trusted software

May perform functions that could indirectly enable a policy to be violated

Formal Verification of Trusted Software

CORRESPONDENCE PROOF CHAIN

POLICY (MODEL)

DESIGN (SPECIFICATION)

IMPLEMENTATION (SOURCE CODE)

MACHINE (OBJECT CODE)

DESIGN VERIFICATION

CODE VERIFICATION

OBJECT CODE AND H/W ANALYSIS
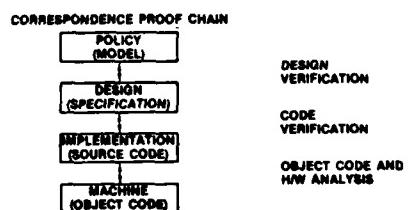
Trusted Software Organization: Security Kernel Approach

Includes a kernel and other trusted software

Kernel acts as a primitive OS

Formal Specification in Software Development
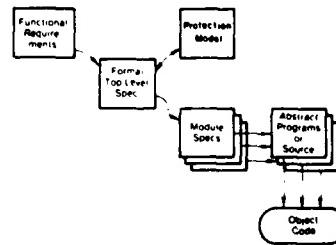
Functional Requirements

Protection Model

Formal Top Level Spec

Module Specs

Abstract Programs or Source

Object Code

Summary

TCBs and operating systems

Provides a verifiable protection base for an operating system

Defining a protection policy

Acts according to a precisely stated protection policy

Developing TCB software

Carries out certain relevant operating system functions

Designed in methodical steps

May be organized as a kernel and other trusted software

TCB Design Lectures

Software interface functions

Human interface functions

Software Interface Functions

John P. L. Woodward

The MITRE
Corporation

General Purpose Timesharing OS

Basic Resources Supported

Processes
  At least one per user
  Some have extraordinary privilege

File System
  Directory organization
  Disk resident
  Media not mountable

I/O devices
  Tapes
  Terminals
  Line printers

TCB Software Interface Functions

General design constraints

Context: general purpose timesharing OS

TLS, verification, and types of channels

Major functional areas

General Purpose Timesharing OS

Security policy enforced

Lattice model applied to subjects and objects

Subjects	Objects
Users	Files (directories, etc)
Processes	I/O devices
	Processes

Security level: (classification, category set)

Policy Rules: Simple security condition
              * - Property
              Tranquility
              Activity
              Erasure

Processes can have privileges to violate policy rules
or perform special functions

Auditing

General Design Constraints

Hardware

Target OS

Performance constraints

"Degree" of security

Verification constraints

Generality

Ease of use

General Purpose Timesharing OS

Security policy rules

Simple security condition:	A subject can read an object iff SL(subject) ≥ SL(object)
* - Property	A subject can write an object iff SL(object) ≥ SL(subject)
Activity	Only objects that are active (that exist) can be accessed.
Tranquility	The SL of active objects cannot change
Erasure	When an object becomes inactive (is deleted) its contents must be erased

Top Level Specification

Information Flow Security Verification

1. Identify objects and "assign" security levels

2. Identify information flow and generate condition tables

3. Generate security inequalities (LEMMAS)

4. Prove that LEMMAS are satisfied

Non-Procedural Specs

O-function exchange (A, B)

Effects

'A = B
'B = A

Flow Table for IPC-Send

Condition	Flow From	Flow To
True	Parameters SL(sender)	IPC-MSG(receiver) SL(receiver)

Lemmas:

SL(sender) · SL(receiver) ◄── Not true

Example Top Level Specification

O-Function IPC-Send(receiver, msg) [sender]
Effects
IPC-MSG(receiver) = msg

Object	Security Level
Parameters	SL(sender)
IPC-MSG(receiver)	SL(receiver)

Secure IPC-Send Specification

O-Function IPC-Send(receiver, msg)[sender]
Exceptions
~SL(sender) < SL(receiver)
Effects
'IPC-MSG(receiver) = msg

Object	Security Level
Parameters	SL(sender)
IPC-MSG(receiver)	SL(receiver)
Exception-Status	SL(sender)
SL(sender)	Low
SL(receiver)	Low

New IPC-Send Specification

```
O-Function IPC-Send(receiver, msg)[sender]
Exceptions
    ~ SL(sender) · SL(receiver)
    · IPC-MSG(receiver) = NULL
Effects
    'IPC-MSG(receiver) = msg
```

Object	Security Level
Parameters	SL(sender)
IPC-MSG(receiver)	SL(receiver)
Exception-Status	SL(sender)
SL(sender)	Low
SL(receiver)	Low
NULL	Low

Flow Table for New IPC-Send

Condition	Flow From	Flow To
SL(sender) · SL(receiver) and IPC-MSG(receiver) = NULL	Parameters SL(sender) SL(sender) Low SL(receiver) Low NULL Low IPC-MSG(receiver) SL(receiver)	Exception-Status SL(sender)

Lemmas
SL(sender) · SL(sender)
Low · SL(sender)
SL(receiver) · SL(sender) ← Not true

Flow Table for Secure IPC-Send

Condition	Flow From	Flow To
SL(sender) · SL(receiver)	Parameters SL(sender) SL(sender) Low SL(receiver) Low	Exception-Status SL(sender)

Lemmas:
SL(sender) · SL(sender)
Low · SL(sender)

Solutions to IPC-Send Insecurity

Allow sending messages only to processes at the same level

Build 2nd exception into an IF statement in EFFECTS:
```
O-Function IPC-Send(receiver, msg)[sender]
Exceptions
    ~ SL(sender) ≤ SL(receiver)
Effects
    if IPC-MSG (receiver) =NULL then 'IPC-MSG (receiver)
        = msg
```
Block caller until msg can be sent

Acknowledge channel
  Delay before returning and audit

Flow Table for Secure IPC-Send

Condition	Flow From	Flow to
SL(sender) · SL(receiver)	Parameters SL(sender) SL(sender) Low SL(receiver) Low	Exception-Status SL(sender) IPC-MSG(receiver) SL(receiver)

Lemmas:
SL(sender) · SL(sender)
SL(sender) · SL(receiver)
Low · SL(sender)
Low · SL(receiver)

Major Functional Areas

Process management

File management

Device management

Miscellaneous functions

Trusted OS Developments

MITRE brassboard 11/45 kernel

MITRE/Honeywell secure MULTICS (Project Guardian)

UCLA data secure UNIX prototype

MITRE Secure UNIX prototype

SDC KVM/370

FACC KSOS-11

Honeywell KSOS/6 (SCOMP)

Major Functional Areas

Process management

File management

Device management

Miscellaneous functions

Typical TCB Domain Architecture

	UNTRUSTED PROCESSES	TRUSTED PROCESSES
USER PROGRAM DOMAIN		
OS EMULATOR DOMAIN		
KERNEL DOMAIN		
HARDWARE		
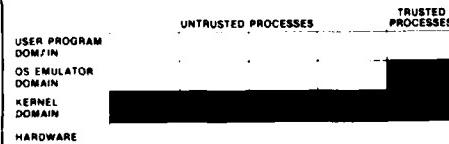
Process Management

Creates the abstraction of processes

Supports the object type: process

Types of processes
  Unprivileged
  Privileged

Types of operations on processes
  Process creation/deletion/status
  Process virtual memory management
  Process switching/scheduling
  Inter-process communication (IPC)

Basic TCB Architecture Options

Internally multiprogrammed kernel

Internally multiprogrammed kernel and trusted processes

Non-interruptible kernel and trusted processes

Process Privilege

Examples
  Violate security model rules
  Call certain TCB functions
  Change certain object attributes
  Set time-of-day clock
  Grant and revoke privileges

Principle of least privilege

Privilege => trust => verification

Process Management

Process creation / deletion / status
  Process-create (context, privileges, level) — process-name
  Process-delete (process-name)
  Process-set-attributes (process-name, privileges, level, other)
  Process-get-attributes (process-name) — (privileges, level, other)

Process virtual memory management

Process switching scheduling

Inter-process communication (IPC)

Process-Create: Example Specification

O-Function process-create(context, privileges, level) (caller) — process-name

Exceptions
  -level $\geq$ 'PT(caller).level
  -privileges $\subseteq$ 'PT(caller).privileges
  '#-of-processes = max-#-of-processes

Effect
  For some x such that 'PT(x).exists = false

    process-name = x
    'PT(x).exists = TRUE
    'PT(x).level = level
    'PT(x).privileges = privileges
    'PT(x).context = context
    '#-of-processes = #-of-processes - 1

Process Attributes

Context

Privileges

Level

Name

Others: Timer, etc

The Name Space Problem

| SECRET NAME: 1 | CREATES | SECRET NAME: 3 |

| UNCLASS NAME: 2 | CREATES | UNCLASS NAME: 4 |
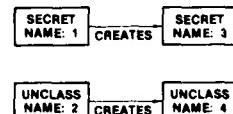
Process-Create: Initial Context, Privileges, Level

Context	Privileges	Level
Same as creator	Subset of creator	Creator
Arbitrary, specified by creator	Subset of creator	Creator
Established, chosen by creator	Any (securely) associated with established context	Creator

---

Kernel Vs. TP Tradeoffs

	Kernel	TP
Option 1	Support fixed # Process Bodies	Handles Create Delete
Option 2	Handles Create Delete w/ Fixed Maximum # Process	N A

Process Management

Process creation deletion status

Process virtual memory management
  Segment-create (level, size, type) — segment-name
  Segment-delete (segment-name)
  Segment-set-attributes (segment-name, level, size, type)
  Segment-get-attributes (segment-name) — (level, size, type)
  Segment-map (segment-name, address, mode)
  Segment-unmap (address)

Process switching scheduling

Inter-process communication (IPC)

Process Management

Process creation/deletion/status

Process virtual memory management

Process switching/scheduling
  When to switch
  Who to switch to
  Mechanics of switching

Inter-process communication (IPC)

Segments

Process address space pieces

Fixed or variable sized

Attributes
  Name
  Level
  Size
  Type: sharable, locked, wired

Kernel Vs. TP Options

	Kernel	TP
Option 1	Supports "switch to process X" function	Determines X, calls kernel
Option 2	Support complex Algorithm to determine who to switch to	N A
Option 3	Supports simple scheduling Algorithm with externally set parameters	Periodically sets parameters (e g Advisory priorities)

Segments: Implementation Issues

Swapping versus paging

Memory swapping device management
  Variable versus fixed sized memory allocation
  Swapping process as a unit: precludes sharing
  Invisible page fault handling
  Page migration

Process Management

Process creation/deletion status

Process virtual memory management

Process switching/scheduling

Inter-process communication (IPC)
  Message size
  Speed of communication
  Security issues

Inter-Process Communication Functions

Message functions
IPC-Send(process, msg)   send message to another process
IPC-Inquire( )   check if any messages to receive
IPC-Receive( ) — msg   receive a message
IPC-Wait( )   Wait for a message

Semaphore functions
IPC-P(semaphore)   wait on semaphore
IPC-V(semaphore)   post semaphore

Software interrupt functions
IPC-Interrupt(channel)   send an interrupt on a
                         particular channel
IPC-Allow(channel)   allow ints on a channel
IPC-Ignore(channel)   ignore interrupts on a channel
IPC-Receive(channel)   receiving interrupt is implicit

Sharing portions of virtual memory among processes
 Segment functions — set up shared memory
 Above functions — synchronization

Directories of Files
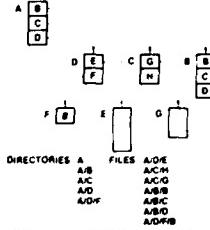
DIRECTORIES A   FILES A/D/E
            A/B   A/C/H
            A/C   A/C/G
            A/D   A/B/B
            A/D/F   A/B/C
                      A/B/D
                      A/D/F/B

Major Functional Areas

Process management

File management

Device management

Miscellaneous functions

File Management

OS goal: directories of files

Security policy options
  Granularity of protection
    Physical / logical disk
    Directory
    File
    Pieces of files
    Records / fields

TCB goal: protect info in files

TCB versus non-TCB options

Kernel versus TP options

File Management

OS goal: directories of files

Security policy options

TCB Goal: protect info in files

TCB versus non-TCB options

Kernel versus TP options

Example functions

File Management

OS goal: directories of files

Security policy options

TCB goal: protect info in files

TCB versus non-TCB options

Kernel versus TP options

Example functions

P-7

File Management

OS goal: directories of files

Security policy options

TCB goal: protect info in files

TCB versus non-TCB options

Kernel versus TP options

Example functions

Files in TCB: Kernel Vs. TP Options

	Kernel	TP
Option 1	Support protected flat file system	N/A
Option 2	Support unprotected pieces of files	Support protected flat file system

---

TCB Vs. Non-TCB Options

	Inside TCB	Outside TCB
Option 1	Support protected directories of files; caller specifies pathname to access files	N/A
Option 2	Support protected flat file system; caller specifies filename to access	Support directories
Option 3	Support protected pieces of files; callers specifies name of piece to access parts of file	Support files and directories

---

File Management

OS goal: directories of files

Security policy options

TCB goal: protect info in files

TCB vs. non-TCB options

Kernel vs. TP options

Example functions
   File-create (level) → file-name
   File-delete (file-name)
   File-open (file-name, mode) → file-descriptor
   File-read (file-descriptor, addr, amount) → data
   File-write (file-descriptor, addr, amount, data)
   File-set-attributes (file-descriptor, level, times)
   File-get-attributes (file-descriptor) → (level, times)

File Management

OS goal: directories of files

Security policy options

TCB goal: protect info in files

TCB versus non-TCB options

Kernel versus TP options

Example functions

File Attributes

Name

Level

Size

Times: last read; last modified

File-Open: Example Specification

O-Function file-open (file-name, mode)[caller] → file-descriptor

Exceptions
FT(file-name).exists = false
¬(FT(file-name).level ≤ PT(caller).level) and read ≤ mode
¬(PT(caller).level ≤ FT(file-name).level) and write ≤ mode

Effects
(Assign new file descriptor that associates
the given file with the requested access mode)

Role of TCB in I/O

If hardware provides...	The TCB must...	
	Non-DMA	DMA
No I/O access control	Interpret all I/O	Interpret all I/O
Access control to devices	Setup initial binding between process and device	Interpret all I/O
Access control to devices and control of device's access to memory	Setup initial binding between process and device	Setup initial binding between process and device

Major Functional Areas

Process management

File management

Device management

Miscellaneous functions

Device Management

Security issues

Kernel vs. TP vs. Untrusted software options

Example functions

Specific I/O device issues

Device Management

Security issues
Non DMA devices: process::device
DMA devices: process::device
device::memory
Role of TCB

Kernel versus TP versus untrusted software options

Example functions

Specific I/O device issues

Kernel Vs. TP Vs. Untrusted Software Options

		Kernel	TP	Untrusted Process
Option 1		Starts I/O; Interrupts TP when done	Contains bulk of Driver Logic; uses Kernel to do I/O	N/A
Option 2		Contains bulk of driver logic; fields interrupts	N/A	N/A
Option 3		Binds device to Process, reflects interrupts to process	N/A	Contains bulk of driver logic; does I/O directly with no Kernel interpretation

Device Management

Security Issues

Kernel versus TP versus untrusted software options

Example functions
  IO-read (device, amount) — data
  IO-write (device, amount, data)
  IO-function (device, function, data)
  IO-set-attributes (device, level)
  IO-get-attributes (device) — level

  Or treat devices as files

Specific I/O device issues

Terminal I/O

Terminal level changing

Terminal assignment

TCB support can be cumbersome
  Character echoing
  Buffering
  Erase and kill processing

Trusted path
  Trusted communication: human — trusted software
  Untrusted software: no access
  Human request via special key on terminal

Device Management

Security issues

Kernel versus TP versus untrusted software options

Example functions

Specific I/O device issues
  Disk
  Tape
  Terminal
  Line printer

Line Printer I/O

Changing level, assignment may be inappropriate

Spooling
  Security requirements:
    1. Files · printer
    2. Level of files unspoofably marked

Tape I/O

Drive level changing

Drive assignment

TCB support straightforward

Major Functional Areas

Process management

File management

Device management

Miscellaneous functions
  Clock/timer I/O
  Auditing

Miscellaneous Functions

Clock/timer I/O
  Not necessarily security relevant
  Virtualized by TCB as I/O devices
  Provided as part of process status
  Separate TCB calls

Auditing

Review of Major Issues

Principle of least privilege

Association between context and privilege

Argument validation

Object names

Exception return channels

Resource exhaustion: quotas

Miscellaneous Functions

Clock/timer I/O

Auditing
  TCB records certain events
    Logins/logouts
    Confinement violations
    Accesses that fail
    Other security-critical events

  Problem: how record with least mechanism

  Solution: audit notification — special process

TCB Software Interface Functions

General design constraints
Context: General purpose timesharing OS
TLS, verification, and types of channels
Major functional areas
  Process management
  File management
  Device Management
  Miscellaneous functions

G. H. Nibaldi

The MITRE
Corporation

Human Interface Functions

User services
  Login logout
  Change user protection
  Change object protection
System security administration
  User control
  Process privilege control
System operations
  Startup
  Shutdown
File system maintenance
  File system consistency checking/repair
  Backup/restore

TCB Organization

                                          HUMAN
                                          INTERFACE
  USER SOFTWARE   OS   TCB
                      UTILITIES UTILITIES

OS SOFTWARE NOT PROTECTION RELATED

                                          TCB SVC
                                          INTERFACE
                TCB

              HARDWARE

Human Interface Functions

User Services
  Login Logout
  Change user protection
  Change object protection

System Security Administration

System Operations

System Maintenance
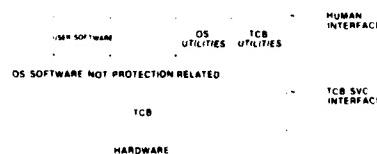
TCB Functionality

Establishes a secure state

Controls state transitions

Binds secure system to external environment

Login

Binds users and programs to processes

FORM	FUNCTIONS	ADVANTAGES	DISADVANTAGES
Account	Predetermined Login	Automatic	Static
	Authenticates user		Trusted path needs access data
TP	Starts process at TP	Dynamic	Must create process at current level
	Interprets input		Must change its owner
	Interpret input		
Login TPs	Authenticate user	Simplicity	Same as above
	Start process		
	Options		
	Allow defaults for Login information		
	Authentication method		
	Degree of user interactions		

Trusted Path

Login/Logout
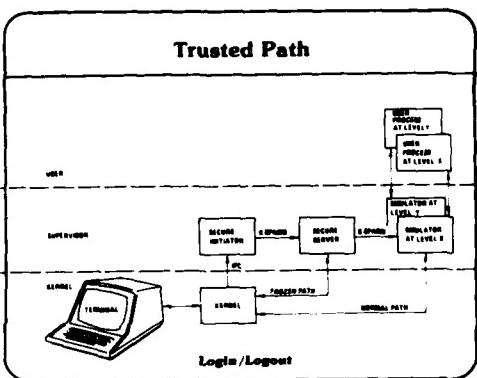
Change Object Protection

Change the protection attributes of an object

FORM	FUNCTION	ADVANTAGES	DISADVANTAGES
Kernel	Change	Efficiency	Complexity
Untrusted Process	Upgrade	Verification	Very restrictive Violates integrity
TP	Change	Flexibility	Privilege to change status May require viewing Synchronization

Options
    Copy object and delete

Logout

Unbind users and processes

FORM	FUNCTION	ADVANTAGES	DISADVANTAGES
Kernel	Stop processes Remove access to tty	Efficiency	Complexity
TP	Stop processes Remove access to tty	Flexibility	Needs process ids Must be able to kill processes Must change owner of tty

Options
    Leave "background" or "detached" jobs
    Operator may logout others

Human Interface Functions

User Services

System Security Administration
    User control
        Process privilege control

System Operations

System Maintenance

Change User Protection

Change the security level of a user

FORM	FUNCTION	ADVANTAGES	DISADVANTAGES
Kernel	Change user to new level	Efficiency	Complexity
TP	Change user to new level	Flexibility	Needs access to authentication data Needs privilege to change attributes Tranquility

Options
    Could be done by logout login
    Range may be mandated
    Old processes may be halted or temporarily suspended
    More authentication data (e g password) may be needed

User Control

Authorize system users

FORM	FUNCTION	ADVANTAGES	DISADVANTAGES
Kernel	Users specified at load time	Efficiency	Data base created off-line
TP	User control editor	Flexibility	Complexity of an editor Access to data base
Untrusted process	User control editor	Verification	Requires confirmation Needs access to data base

Process Privilege Control

Associate process privileges with programs

FORM	FUNCTION	ADVANTAGES	DISADVANTAGES
Kernel	Fix privileges at load time	Simplicity	Lacks flexibility
TP	Privilege control editor	Flexibility	Flexibility Needs privilege to set privileges Must access control data
Untrusted process	Privilege control editor	Verification	Requires confirmation Must access control data

Shutdown

Halt system operations

FORM	FUNCTION	ADVANTAGES	DISADVANTAGES
Kernel	Stop processes Sync filesystem	Efficiency	Complexity
TP	Stop processes Sync filesystem	Flexibility	Needs process ids Needs privilege to sync filesystems

Options
    May preserve checkpoint data

Human Interface Functions

User Services

System Security Administration

System Operations
    Startup
    Shutdown

System Maintenance

Human Interface Functions

User Services

System Security Administration

System Operations

System Maintenance
    File System Consistency Checking / Repair
    Backup / Restore

Startup

Establish the initial secure state for TCB

FORM	FUNCTION	ADVANTAGES	DISADVANTAGES
Boot	Initialize TCB	Efficiency	Complexity
Stand-alone Program	Initialize TCB	Simplicity	
Kernel	Initialize TCB	Efficiency	Complexity

Options
    Set up an initial process
    Recover from checkpointed operation

File System Consistency Checking / Repair

Enable checkout and repair of ailing filesystems

FORM	FUNCTION	ADVANTAGES	DISADVANTAGES
Stand-alone	Check Repair	Dedicated machine No sync problems	Dedicated machine Separate drivers Separate interface
Kernel	Check Repair	Efficiency	Complexity
TP	Check Repair	Kernel I O	Needs kernel object for filesystem
2 TPs	Check Repair	Flexibility and simplicity	Same as above

Backup/Restore

Allows for the capture of files and their subsequent restoration

FORM	FUNCTION	ADVANTAGES	DISADVANTAGES
Stand-alone	Volume copy	Efficiency	Separate drivers
			Dedicated machine
Kernel	Volume copy	Efficiency	Complexity
	Auto backups	Disk mount	High CPU overhead
	on updates	CPU overhead	
Untrusted Process	Single level volume copy	Verification	Single level copy
			Needs kernel object for filesystem
			Needs access to storage device
TP	Multilevel volume copy	Efficiency	Needs kernel object for filesystem
	File copy	Uses kernel objects	Needs to know labels
			Needs access tables
			Needs access to storage device
TPs	Backup	Simplicity	Saves no objects
	Restore		

Human Interface Functions

User services
   Login logout
   Change user protection
   Change object protection

System security administration
   User control
   Process privilege control

System operations
   Startup
   Shutdown

File system maintenance
   File system consistency checking repair
   Backup restore

KSOS:
AN EXAMPLE OF A TRUSTED COMPUTING BASE

Dr. E.J. McCauley

Ford Aerospace &
Communications Corporation

KSOS STATUS

* Primitive kernel operational

* Skeletal emulator running on simulated kernel

* Support NKSR operational

* New kernel and NKSR formal specs

* Advanced concepts in testing

Ford Aerospace &
Communications Corporation

TOPICS

* KSOS Design Choices

* KSOS Security Assurance

* Insights into TCB Design and Implementation

* Hindsight

Ford Aerospace &
Communications Corporation

KSOS DESIGN CHOICES

Ford Aerospace &
Communications Corporation

KSOS DESIGN CHOICES

* Larger, more monolithic Kernel

* Swapping vs. Demand Paging

* Secure Path, Secure Server

* Type Extension

* Protection Domain Modification

* Network Interface Architecture

* Auditing

KSOS Kernel Objects

Processes	Program in execution
Process Segments	Portions of virtual memory of a process
Files	Linear array of data blocks, "flat" name space
Devices	Special type of file
Subtypes	Encapsulation tool

Ford Aerospace &
Communications Corporation

Kernel Objects

Every object has

A Name (Secure Entity Identifier "SEID")

Type Independent Information

Owner (user and group)

Security classification (e.g. TOP SECRET)

Security compartment set (e.g. NOFORN caveats)

Integrity classification

Integrity compartment set (now always null)

Discretionary access information

Ford Aerospace &
Communications Corporation

KSOS Kernel Process

- Cheap, plentiful
- May be privileged K_invoke K_spawn
- K_fork "cloning"
- Inter Process Communication

  messages
  shared segments

Ford Aerospace &
Communications Corporation

KSOS Kernel Files and Devices

- "Flat" name space
- Linear array of data blocks
- Single file up to 300 Mbytes (800? 1200?)
- Mountable volumes fully protected

Ford Aerospace &
Communications Corporation

SWAPPING vs DEMAND PAGING

- PDP-11 Memory Management Unit Limitations
- 16 bit virtual address
- Most programs are very small: working set is all of their pages

Ford Aerospace &
Communications Corporation

KSOS Kernel Process Segments

- Variable sized
- Rendezvous with shared segments by names
- Options for system designer

  normal
  sticky
  locked

Ford Aerospace &
Communications Corporation

KSOS Secure Terminal Interface

- Need unspoofable path to secure services

Ford Aerospace &
Communications Corporation

R-2

SECURE FLOW
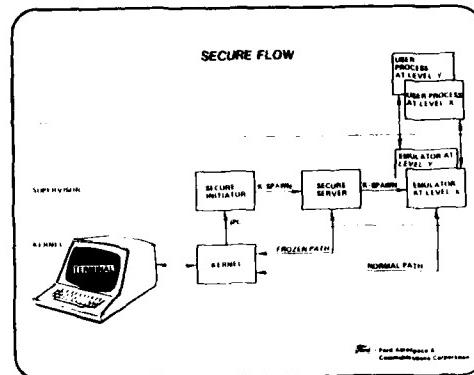
PROTECTION DOMAIN MODIFICATION

- Untrusted process issues K_invoke or K_spawn Kernel call

- Rendezvous with trusted intermediary code segment, privileges set from those of this segment

- Control transferred to intermediary segment

- Intermediary program alters protection environment
  - Initializes segments
  - Sets security level, privileges, etc

- Control transferred to fixed location in altered environment
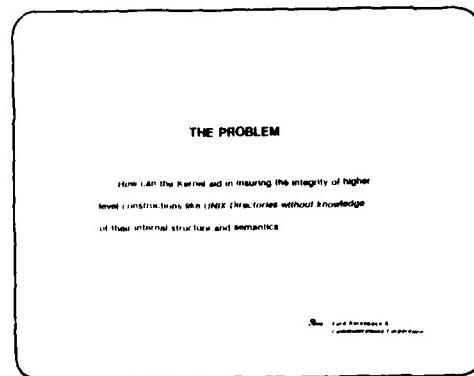
THE PROBLEM

How can the Kernel aid in insuring the integrity of higher level constructions like UNIX Directories without knowledge of their internal structure and semantics
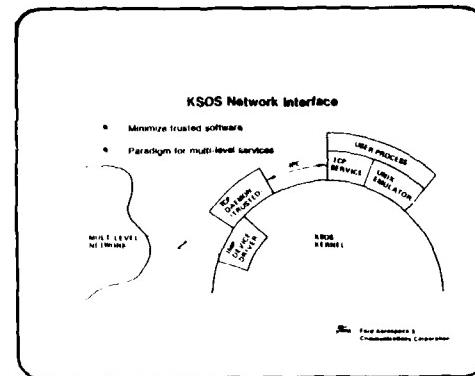
KSOS Network Interface

- Minimize trusted software
- Paradigm for multi-level services
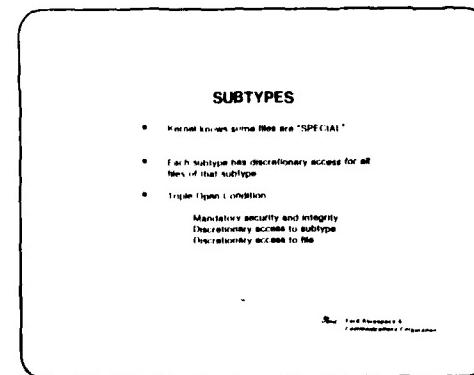
SUBTYPES

- Kernel knows some files are "SPECIAL"

- Each subtype has discretionary access for all files of that subtype

- Triple Open Condition
  - Mandatory security and integrity
  - Discretionary access to subtype
  - Discretionary access to file

Auditing in KSOS

R-3
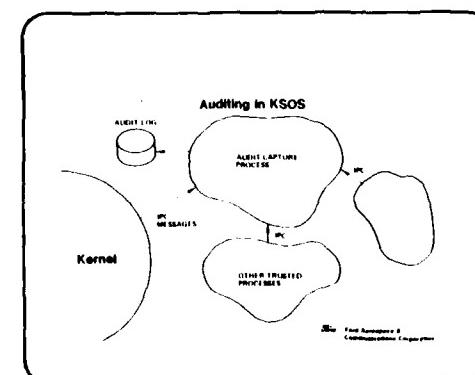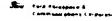
KSOS SECURITY ASSURANCE

INSIGHTS INTO TCB DESIGN

KSOS SECURITY ASSURANCE
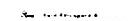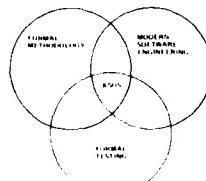
INSIGHTS INTO TCB DESIGN AND IMPLEMENTATION

* It can be done!

* Need for consistency between different languages care in their use

* Utility and benefits of formal specifications

* Code proofs are not yet practical except for demonstrations However, being ready to do them is of great benefit

* Need for additional tools and concepts

SCHEMA FOR CONSTRUCTION OF PROVABLE SYSTEMS
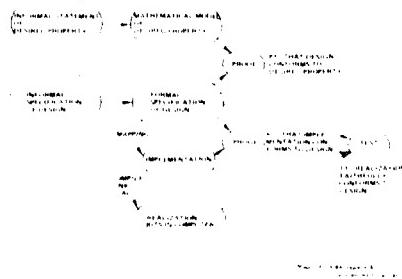
WHAT TO ASK FOR

* Verification plan (P-O system security plan?)

* Top-level (A) spec
  Incorporate mathematical models
  Describe minimum implementation standards

* Development specs (B's)
  Major interfaces
  Formal specs for trusted components

* Product specs (C's)
  Detailed design
  Low level formal specs for procedures to be proven

* Verification reports

* Vulnerabilities analysis

R-4

WHAT THE IMPLEMENTOR SHOULD HAVE

- Formal spec tools
  - Spec language (SPECIAL, INA JO, GYPSY, DREAM)
  - Spec proof tools

- Verifiable language + language support(?)
  - MODULA
  - PASCAL
  - GYPSY
  - ADA

- Code proof tools

- Mathematically inclined personnel and understanding management

- Disciplined implementation environment
  - SCCS or equivalent
  - Design language
  - System builder
  - Test generation tools

- KSOS itself uses PWB/UNIX (tm) tools for system generation

HINDSIGHT · THINGS THAT WORKED

- Success of disciplined methodology

- Value of formal specifications for unexpected purposes

- Integrated development environment worked well

- Personnel accepted formal methods easily

- Although occasionally annoying MIL-SPEC documentation was useful

- Having a model to work against very helpful

HINDSIGHT

HINDSIGHT · WHAT MIGHT HAVE BEEN DONE BETTER

- Better integration of segment and file systems

- More insight into consistency between multiple representations

- Better implementation language

- Simpler secure path mechanism

- Alternate Emulator structure

SECURE COMMUNICATIONS PROCESSOR
SCOMP
&
KERNELIZED SECURE OPERATING SYSTEM
(KSOS 6)

CHARLES H. BONNEAU
HONEYWELL, INC
ST. PETERSBURG, FLORIDA 33733

SECURITY PROTECTION MODULE FEATURES

- FAST PROCESS SWITCHING
  - PROCESS DESCRIPTOR TREE DEFINITION VIA DESCRIPTOR
    BASE ROOT
  - AUTOMATIC LOAD OF DESCRIPTORS
- 4 LEVEL MEMORY DESCRIPTOR SYSTEM
  - CAN BE ENTRY AT ANY LEVEL
  - ELEMENT SIZE WORDS 512
  - TABLE 128 WORDS
- ACCESS MEDIATION
  - DEVICE TO DEVICE
  - DEVICE TO MEMORY
- MULTIPLE LEVEL RING STRUCTURE
  - 2 PRIVILEGED, 2 NON-PRIVILEGED RINGS
  - READ, WRITE, EXECUTE, AND CALL BRACKETS
  - RING CROSSING SUPPORT INSTRUCTIONS
- PAGE FAULT RECOVERY SUPPORT

OBJECTIVES

- ADD PROTECTION HARDWARE TO COMMERCIAL MACHINE WHICH MAKES
  IT EASIER TO BUILD SECURE SYSTEMS

- DEVELOP TRUSTED SOFTWARE
  - ENFORCE DOD SECURITY POLICY
  - FORMALLY VERIFIABLE
  - SUPPORT MANY OTHER APPLICATIONS

SYSTEM CHOICES

- FILES VERSUS SEGMENTS

- PAGING VERSUS SWAPPING

- KERNEL INTERRUPTIBILITY

- ACCESS CONTROL ATTRIBUTES MODIFICATION

- TRUSTED SOFTWARE VERSUS KERNEL GATE
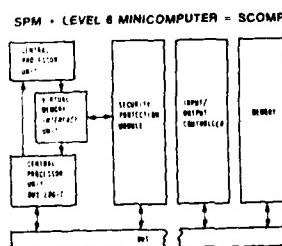
SPM + LEVEL 6 MINICOMPUTER = SCOMP

SYSTEM CONSIDERATIONS

- EXTERNAL REQUIREMENTS

- SOFTWARE SUPPORT

- KERNEL COMPLEXITY

- PERFORMANCE

SYSTEM DESIGN

- NON-FILESYSTEM IS OUTSIDE KERNEL
- FILES CONSTRUCTED EXTERNALLY USING SEGMENTS
- DEMAND PAGING VIRTUAL MEMORY
- NON-DISCRETIONARY ACCESS CONTROL - BELL AND LAPADULA
  - PRIVILEGE
  - ALERT ATTRIBUTES NOT FIXED
- DISCRETIONARY ACCESS CONTROL
  - UNIX R,W,E FOR OWNER, GROUP, OTHER
  - RING BRACKETS FOR OWNER, GROUP, OTHER
  - SUBTYPES
- KERNEL INTERRUPTIBILITY
  - KERNEL OPERATIONS MAY BLOCK
  - KERNEL OPERATIONS NOT INTERRUPTED
    - NO PROCESS SWITCH
    - SEGMENT ACCESS RECHECK

PROCESS STATUS INFORMATION

- ADVISORY PRIORITY
- PRIVILEGE INDICATOR
- ... INTERRUPT ENABLE INDICATOR
- ACCESSIBLE OBJECT SUBTYPES
- PRIVILEGES
- RING 2/3 EXECUTION TIME
- RING 2/3 STACK ADDRESS

SYSTEM DESIGN (CONTINUED)

- INFORMATION CHANNEL CONTROL
  - UPGRADED ARGUMENT
    - READABILITY DETERMINES RESPONSE
    - ... HIGH GARBAGE CAN SEGMENT

PROCESS PRIVILEGES

- MODIFY PRIVILEGE
- SET LEVEL
- SET ... LEVEL
- SET DISCRETIONARY ACCESS
- SET USER GROUP
- SET SUBTYPE ACCESS
- SET OBJECT STATUS
- TERMINAL ...
- ... ACCESS CONTROL
- VIOLATE SIMPLE SECURITY
- VIOLATE SECURITY * PROPERTY
- VIOLATE SIMPLE INTEGRITY
- VIOLATE INTEGRITY * PROPERTY
- VIOLATE DISCRETIONARY ACCESS

KERNEL OBJECTS

KERNEL OBJECT MANIPULATION

- CREATE UPGRADED OBJECTS
- CREATE OBJECT AT PROCESS LEVEL ONLY
- ... UPGRADED SEGMENT
- ... AT PROCESS LEVEL ONLY
- SET SEGMENT AT PROCESS LEVEL
- ... NON-DISCRETIONARY ONLY
- SET OBJECT FUNCTIONS
  - DISCRETIONARY + NON-DISCRETIONARY HELD

VISIBLE FUNCTIONS

PROCESSES:	SEGMENTS:	DEVICES:
● CREATE PROCESS	● CREATE SEGMENT	● CREATE DEVICE
● DELETE PROCESS	● DELETE SEGMENT	● REMOVE DEVICE
● RELEASE PROCESS	● GET SEGMENT ACCESS	● GET DEVICE ACCESS
● GET PROCESS ACCESS	● SET SEGMENT ACCESS	● SET DEVICE ACCESS
● SET PROCESS ACCESS	● GET SEGMENT STATUS	● GET DEVICE STATUS
● GET PROCESS STATUS	● SET SEGMENT STATUS	● SET DEVICE STATUS
● SET PROCESS STATUS	● MAP SEGMENT	● MAP DEVICE
● SET PROCESS SUBTYPES	● UNMAP SEGMENT	● UNMAP DEVICE
● RECEIVE MESSAGE	● WIRE SEGMENT	● SECURE TERMINAL LOCK
● SEND MESSAGE	● UNWIRE SEGMENT	● SECURE TERMINAL UNLOCK
● INTERRUPT RETURN	● SYNC SEGMENT	● MOUNT
		● UNMOUNT
		● READ REAL CLOCK

KERNEL ARCHITECTURE

● DISTRIBUTED

  NO CONTEXT SWITCH TO PERFORM KERNEL FUNCTION

  - LARGE VIRTUAL ADDRESS SPACE

● KERNEL PROCESSES

  - PROC 0 - SCHEDULER: SERVICE KILL PROCESS SIGNALS

  - PROC 1 - AUDITOR   GENERATE AUDIT SEGMENTS

  - PROC 2 - SEGMENT RELEASER: RELEASE INACTIVE SEGMENTS

  - PROC 3 - SYSTEM SYNC   SYNC ALL MOUNTED FILESYSTEMS

KERNEL INTERNAL FUNCTIONS

● RESOURCE MANAGEMENT

  - SCHEDULER
  - MEMORY MANAGEMENT

● TRAP MANAGEMENT

● INTERRUPT MANAGEMENT

KSOS STRUCTURE

RING 0   SECURITY KERNEL

RING 1   UNIX EMULATOR
          TRUSTED SOFTWARE

RING 3   USER PROGRAMS

KERNEL TRUSTED SOFTWARE INTERFACES

● SECURE INITIATOR

● SECURE AUDITOR

● OPERATOR INTERFACE

ADDRESS SPACE PARTITIONING

MEMORY

● SEG ... KERNEL GLOBAL

● SEG ... KERNEL LOCAL

● SEG ... USER LOCAL

    ... EMULATOR

    SEG ... USER

DEVICE

● DEV 0-31   KERNEL GLOBAL

● DEV ... USER LOCAL

S-3

DESIGN CHOICES

- DESIGN PARAMETERS
  - KERNEL FUNCTION CALL BRACKETS
  - NUMBER OF PAGES PER BLOCK
- GLOBAL MEMORY MODEL
- ...
- ... AND ...
- ... MANAGEMENT OF KERNEL RESOURCE
  - AMOUNT OF LOCKED MEMORY PER PROCESS SIGNIFICANT
  - VERIFICATION ISSUE

VERIFICATION RESULTS/CONCLUSIONS

- CURRENT TLS PROVABLE, BUT
  - NUMBER OF FORMULAS PRODUCED REQUIRES UNACCEPTABLE DEC 10-KL CPU TIME BY THEOREM PROVER
- WHY
  - AMOUNT OF DETAIL IN INDIVIDUAL MODULES
  - MODULE ORGANIZATION
- TLS BEING REWRITTEN TO ABSTRACT OUT AND/OR CONSOLIDATE INFORMATION TO ACHIEVE A SMALLER, BETTER ORGANIZED TOP LEVEL SPECIFICATION WHICH IS EQUIVALENT IN INFORMATION CONTENT TO CURRENT TLS
- RESULTS OF REWRITE ENCOURAGING
  - REWRITE AND REORGANIZATION OF LOWER B MODULES REDUCED NO. OF GENERATED FORMULAS FROM THE DEVICES MODULE BY APPROXIMATELY 50 PERCENT
- NEED
  - DEFINITION OF LEVEL OF DETAIL IN TLS
  - SPECIFICATION STYLE RULES

... PROOF OF CORRECTNESS

- USES HIERARCHICAL DEVELOPMENT METHOD (HDM) WHERE
  - FORMALLY STATED REQUIREMENTS = MULTILEVEL SECURITY MODEL
  - FORMAL SPECIFICATIONS DEFINING THE DESIGN - SPECIAL, ...
- FORMAL VERIFICATION OF SYSTEM DESIGN
- ILLUSTRATION PROOF OF IMPLEMENTATION
  - IMPLEMENTATION IN UCLA PASCAL

TRUSTED SOFTWARE

INITIATOR	SPAWN SECURE SERVER (INIT)
SERVER	SPAWN HKSP SERVICES (MINI-SHELL)
LOGIN	SIGN ON AND INITIATE USER ENVIRONMENT (LOGIN)
STTY	SET TYPEWRITER OPTIONS (STTY)
OPERATOR INTERFACE	SUPPORT STARTUP AND CHECK DISK PACKS
STARTUP	BUILD KERNEL AND INITIALIZE SYSTEM (BOOT)
LOGOUT	USER SIGN OFF
CREATE	CREATE PRIVILEGED USER PROCESS
GARBAGE COLLECTOR	DELETE UPGRADED DIRECTORIES
MAP NAME	USER NAME TO KERNEL NAME MAPPING
TERMINAL I/O	TERMINAL DRIVERS TO SUPPORT TRUSTED SOFTWARE
MAKE FILE	BUILD INITIAL FILE SYSTEM (MKFS)
DATA BASE EDITOR	MODIFY KERNEL AND SERVER DATA BASE (MKCONF, CONF)
ACCESS MODIFIER	CHANGE FILE SECURITY LEVEL

KERNEL TLS

MODULE	NO. OF FUNCTIONS
PROCESS VIRTUAL SPACE	44
INTERPROCESS COMMUNICATION	3
PROCESS OPERATORS	9
EVENTS	21
MOUNTABLE FILE SYSTEMS	15
FILES	37
DIRECTORIES	12
ACCESS CONTROL	18
DEVICE DRIVERS	5
PRIVILEGE CONTROL	5
OBJECT ACCESS INFORMATION	19
SYSTEM LEVEL	8
OBJECT NAMES	8
CLOCK	1
	205

- APPROXIMATELY 4,000 LINES OF SPECIAL
- 67 VISIBLE FUNCTIONS { 55 SOFTWARE GATES / 12 HARDWARE GATES

DATA

- HARDWARE
  - ...
- SOFTWARE
  - ...

"Innovation in UCLA Secure
UNIX"

Dr. Gerald Popek
UCLA

ANOTHER FLAW

(from the I've Been Mired Corp op sys)
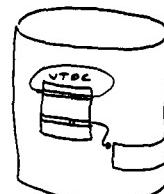
A REAL SECURITY
PENETRATION ROUTE

Computer Memory

User
Memory

Supervisor
Memory
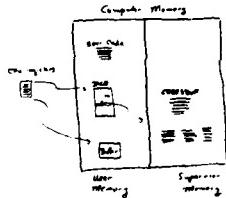
(the Itty Bitty Machine
Corp operating system)

LEVELS OF KERNELS

SECURE NETWORK
OPERATION

User processes

←NM

Kernels

encryption
unit

h/w

h/w

encryption
unit

Secure   Networks

Yodal-eh-ee-hoo...
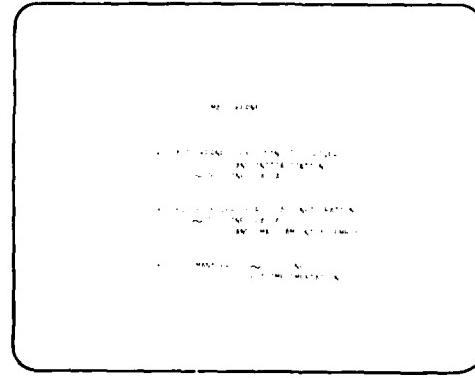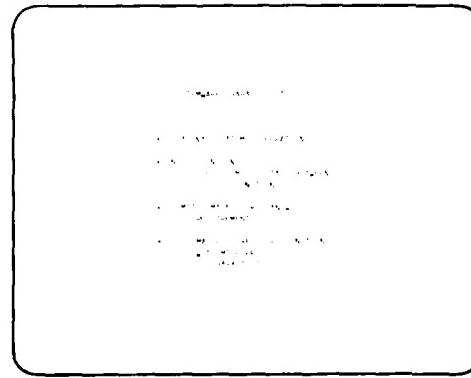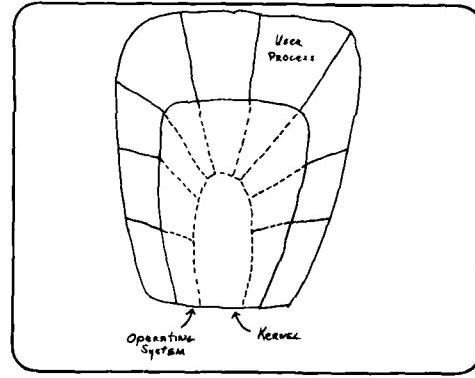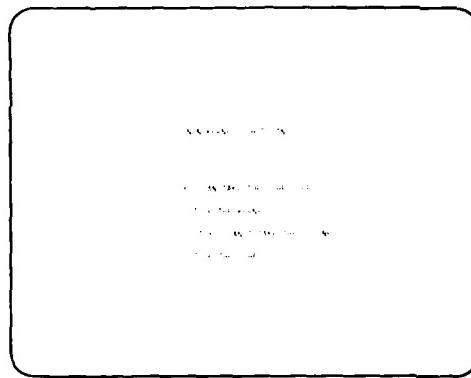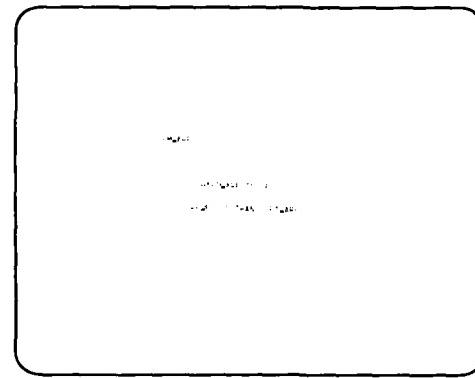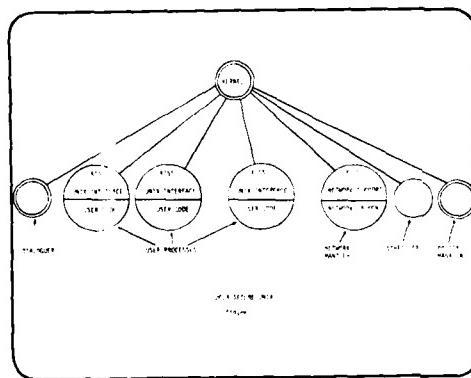
INTER PROCESS COMMUNICATION

INTER PROCESS COMMUNICATION

Kernel   IPC   Data Structure
Rendezvous   Table

PAGING
UCLA KERNEL

CLASSES OF PENETRATIONS ELIMINATED ON VM...

- DATA DEPENDENT VIOLATIONS
    - ...
    - BECAUSE OF PROGRAMS

- ...
    - ...
    - ...
    - ...

DATA IN VM/370

- VM/CP SIMULATES I/O DEVICES FOR THE UNIT
- VM/CP MAPS BETWEEN VIRTUAL DEVICES AND REAL DEVICES
- REAL DISKS ARE PARTITIONED INTO MINIDISKS
- ...
- FILES ARE ... ON THE UNIT

PHYSICAL PACK SHARING

USER B	291 MINIDISK	52 CYLINDERS

TEMPORARY SPACE FOR VM/370 PAGING ...

USER A	...	...
USER C	...	...

CPCMS2

KERNEL CALLS

PAGING

LOOKS FOR PAGE FRAMES IN MEMORY

EMPTY N

FULL FREE

ATTACH 20

ABBREVIATED COMPUTER SECURITY BIBLIOGRAPHY
8 January 1980

GENERAL

Bell, D. E., and LaPadula, L. J., "Secure Computer Systems," ESD-TR-73-278, Volume I-III, The MITRE Corporation, Bedford, MA, November 1973-June 1974.

Bell, D. E., and LaPadula, L. J., "Computer Security Model: Unified Exposition and Multics Interpretation," ESD-TR-75-306, The MITRE Corporation, Bedford, MA, March 1976 (AD 203588).

Lampson, B. W., "A Note on the Confinement Problem," Communications of the ACM, Volume 16, Number 10, October 1973, pp. 613-615.

Lipner, S. B., "Comment on the Confinement Problem," ACM Operating Systems Review, Volume 9, Number 5, May 1975, pp. 192-196.

Linden, T. A., "Operating System Structures to Support Security and Reliable Software," ACM Computing Surveys, Volume 8, Number 4, December 1976, pp. 409-445.

Saltzer, J. H., and Schroeder, M. D., "The Protection of Information in Computer Systems," Proceedings of the IEEE, Volume 63, Number 9, September 1975, pp. 1278-1308.

Smith, L., "Architectures for Secure Computer Systems," ESD-TR-75-51, The MITRE Corporation, Bedford, MA, April 1975 (AD A09221).

Tangney, J. D., "Minicomputer Architectures for Effective Security Kernel Implementations," ESD-TR-78-170, The MITRE Corporation, Bedford, MA, October 1978.

Anderson, J. P., "Computer Security Technology Planning Study," ESD-TR-73-51, Volume I and II, James P. Anderson & Co., Fort Washington, PA, October 1972.

Attanasio, C. R., Markstein, P. W., and Phillips, R. J., "Penetrating an Operating System: A Study of VM/370 Integrity," IBM Systems Journal, Volume 15, Number 1, 1976, pp. 102-116.

Biba, K. J., "Integrity Considerations for Secure Computer Systems," ESD-TR-76-372, Electronic Systems Division, AFSC, Hanscom AFB, MA, April 1977, (AD A039324).

DeWolf, J. B., and Szulewski, P. A., (ed.), "Final Report of the 1979 Summer Study on Air Force Computer Security," R-1326, The Charles Stark Draper Laboratory, Inc., Cambridge, MA, October 1979.

Feiertag, R. J., Levitt, K. N., and Robinson, L., "Proving Multilevel Security of a System Design," Proc. ACM Sixth Symposium on Operating Systems Principles, November 1977, pp. 57-65.

Lee, T. M. P., et al., "Processors, Operating Systems, and Nearby Peripherals: A Consensus Report," Secure Operating System Technology Papers for the Seminar on the DoD Computer Security Initiative Program, NBS Special Publication, Gaithersburg, MD, July 1979.

Lipner, S. B., "Security Considerations in Information System Design," NBS Special Publication 404, Privacy and Security in Computer Systems, September 1974.

Nibaldi, G. H., "Proposed Technical Evaluation Criteria for Trusted Computer Systems," M79-225, The MITRE Corporation, Bedford, MA, October 1979.

Schell, R. R., "Computer Security: The Achilles' Heel of the Electronic Air Force," Air War College Research Report RR-468, Naval Postgraduate School, Monterey, CA, April 1978.

Ware, W. H., (ed.), "Security Controls for Computer Systems," Report of Defense Science Board Task Force on Computer Security, The RAND Corporation, Santa Monica, CA, Reissued October 1979.

SECURE SYSTEMS

McCauley, E. J., and Drongowski, P., "KSOS: Design of a Secure Operating System," Proceedings of 1979 NCC, Volume 48, AFIPS Press, New York, June 1979, pp. 345-354.

Popek, G. J., and Kline, C. S., "Issues in Kernel Design," Proceedings of the 1978 National Computer Conference, Volume 47, AFIPS Press, June 1978, pp. 1079-1086.

Schaefer, M., Gold, B., Linde, R., and Scheid, J., "Program Confinement in KVM/370," Proceedings of 1977 ACM Conference, October 1977, pp. 404-410.

Schell, R. R., Downey, P. J., and Popek, G. J., "Preliminary Notes on the Design of Secure Military Computer Systems," NCI-73-1, Electronic Systems Division (AFSC), L. G. Hanscom Field, Bedford, MA, January 1973.

Schiller, W. L., "The Design and Specification of a Security Kernel for the PDP-11/45," ESD-TR-75-69, The MITRE Corporation, Bedford, MA, May 1975, (AD A011712).

Schiller, W. L., Withington, P. T., and Woodward, J. P. L., "Design and Abstract Specification of a Multics Security Kernel," ESD-TR-77-259, Volumes I and II, The MITRE Corporation, Bedford, MA.

Woodward, J. P. L. and Nibaldi, G. H., "A Kernel-Based Secure UNIX Design," ESD-TR-79, The MITRE Corporation, Bedford, MA, November 1977.

Feiertag, R. J., and Neumann, P. G., "The Foundations of a Provably Secure Operating System (PSOS)," Proceedings of 1979 NCC, Volume 48, AFIPS Press, New York, June, 1979, pp. 329-334.

Gold, B. D., et al., "A Security Retrofit of VM/370," Proceedings of 1979 National Computer Conference, Volume 48, AFIPS Press, New York, 1979, pp. 335-344.

Neumann, P. G., et al., "A Provably Secure Operating System: The System, its Applications, and Proofs," SRI International, Menlo Park, CA, February 1977.

Popek, G. J., et al., "UCLA Secure UNIX," Proceedings of the 1979 NCC, Volume 48, AFIPS Press, New York, June 1979, pp. 355-364.

SPECIFICATION/VERIFICATION

Berson, T. A., et al., "KSOS: Development Methodology for a Secure Operating System," Ford Aerospace and Communications Corporation, Palo Alto, CA.

Robinson, L., Levitt, K. N., Neumann, P. G., and Saxena, A. R., "A Formal Methodology for the Design of Operating System Software," in Current Trends in Programming Methodology, R. T. Yeh ed., Volume 1, Prentice-Hall, Englewood Cliffs, NJ, April 1977.

Robinson, L., Neumann, P. G., Levitt, K. N., and Saxena, A. R., "On Attaining Reliable Software for a Secure Operating System," 1975 International Conference on Reliable Software, Los Angeles, CA, April 1975, pp. 267-284.

DeMillo, R. A., Lipton, R. J., and Perlis, A. J., "Social Processes and Proofs of Theorems and Programs," Communications of ACM, Vol. 22, No. 5, May 1979, pp. 271-280.

Denning, D. E., and Denning, P. J., "Certification of Programs for Secure Information Flow," Communications of the ACM, Volume 20, Number 7, July 1977, pp. 504-513.

Denning, D. E., "A Lattic Model of Secure Information Flow," Communications of the ACM, Volume 19, Number 5, May 1976, pp. 236-243.

Millen, J. K., "Security Kernel Validation in Practice," Communications of the ACM, Volume 19, Number 5, May 1976, pp. 243-250.

Millen, J. K., "Operating System Security Verification," M79-223, The MITRE Corporation, Bedford, MA, September 1979.

Millen, J. K., "Formal Specifications for Security," Symposium Proceeding: Trends and Applications 1977, Computer Security and Integrity, Gaithersburg, MD, May 1977, (IEEE 77 CH 1204-7 C).

Millen, J. K., "Example of a Formal Flow Violation," COMPSAC 78, Chicago, IL, November 1978, (IEEE 78 CH 1338-3 C).

Millen, J. K., "Logical Channel Theory, " MTR-3833, The MITRE Corporation, Bedford, MA, December 1979.

Parnas, D. L., "A Technique for Software Module Specification with Examples," Communications of the ACM, Volume 15, Number 5, May 1972, pp. 330-336.

Popek, G. J., and Farber, D. A., "A Model for Verification of Data Security in Operating Systems," Communications of the ACM, Volume 21, Number 9, September 1978, pp. 737-749.

Walter, K. G., Ogden, W. F., et al., "Initial Structured Specification for an Uncompromisable Computer Security System," ESD-TR-75-82, Case Western Reserve University, Cleveland, OH, July 1975.

APPLICATIONS

Ames, S. R., and Oestreicher, D. R., "Design of a Message Processor System for a Multilevel Secure Environment," Proceedings of 1978 NCC, Vol. 47, Anaheim, CA, June 1978, pp. 765-771.

Padlipski, M. A., et al., "KSOS: Computer Network Applications," Proceedings of 1979 NCC, Vol. 48, AFIPS Press, New York, June 1979, pp. 373-382.

Woodward, J. P. L., "Applications for Multilevel Secure Operating Systems," Proceedings of the 1979 NCC, Vol. 48, AFIPS Press, New York, June 1979, pp. 319-328.

Woodward, J. P. L., "ACCAT Guard System Specification (Type A)," MTR-3634, The MITRE Corporation, Bedford, MA, August 1978.

COMPUTER SECURITY TECHNOLOGY GLOSSARY

Access Level   The combination of the security level and the integrity level of a subject or object.

Activity   The security model rule that states that only active objects can be accessed. Once an object is made inactive, it cannot be accessed unless it is made active again.

Access Control   A strategy for protecting objects from unauthorized access.

Address Space   The virtual memory that can be addressed by a process. The maximum size of a process's address space is usually a function of the underlying hardware.

Attention Character   In TCB design, a character that, if entered from a terminal, tells the TCB that the user wants a secure communications path from the terminal to some trusted code to provide some type of secure service for the user, such as logging in or logging out. Contains authorization information defining how the object may be used (e.g., read, write).

Category Set   A category set, part of a security level, is a list of information categories applicable to an object or subject. Categories correspond roughly to the topic area of the information. A subject must be cleared for all categories on an object to read the object. (See compartment.)

Channel	A means of transferring information from one object to another object.
Classification	An access token usually associated with an information repository, or object, though sometimes associated with subjects also. The classification of a subject or object, along with a category set, makes up the security level of an object.
Clearance	An authorization allowing a person (or his surrogate within a computer, a process) access to classified information. A clearance is represented as a classification in a security level.
Compartment	Compartments are a mutually exclusive way to assign categories. If an object is compartmented, then it generally has only one category, called a compartment. Compartments are used mostly in the intelligence community. They are implemented in a kernel-based system using categories.
Concurrency	The occurrence of two or more operations at the same time. A kernel is said to have concurrency if interrupts are enabled to allow interruption of the execution of kernel operations. Concurrency in a kernel has an adverse effect on the ease of verification of the kernel, but increases the functionality of the kernel.
Confinement	A condition where a process is unable to leak information illegitimately. Because processes must share computer resources, confinement channels are difficult to avoid. Such channels can be exploited to allow unauthorized read access to information. (See channels, legitimate, illegitimate, covert, storage, and timing channels.)

Covert Channel	A confinement channel involving colluding processes. It results from the shared use of common limited computer resources.
Denial of Service	The prevention of authorized access to a computer system.
Discretionary Security	The aspect of security policy policy implementing the "need to know" requirement for access to information.
DoD Security Policy	The complete body of law, regulations and policy concerning the safeguarding of Defense sensitive information. DoD security policy includes all the Espionage laws, the DoD regulations and DoD authorized commercial classification, for handling and access to information concerning national defense. The basic policy set four levels and several categories of non-discretionary information control and requires that anyone accessing classified information have the need to know the particular information in question. A representative description of the policy can be found in AFR 205-1.
Domain	Domains allow hardware features to be restricted or subsetted. For example, DEC PDP 11/45 or 11/70 has three execution domains: kernel, supervisor, and user. The kernel domain is the most privileged, and allows access to all hardware features including memory maps and I/O; the other domains do not allow these privileges, but allow access as controlled by the kernel domain).

Emulator	That portion of a secure, kernel-based system that creates an operating system compatible environment out of the environment provided by the kernel. In the case of KSOS, the emulator maps the kernel environment into the UNIX environment.
Erasure	The security model rules that states that an object must be "erased" before being activated. This means that all objects have a precise, pre-defined value when created (e.g., all zeros).
Euclid	A Pascal-based higher order language designed to facilitate verification.
Finite State Machine	An abstract concept on which a number of protection models are based. In theory, by starting with a secure initial state and following the model rules for all changes, then all states of the machine are secure.
Flow Control	A strategy for protecting the contents of information objects from being transferred to objects at improper access levels.
Formal specifications	See Top-Level Specification.
Granularity	Granularity of protection refers to the size of the smallest protectable unit of information. In a kernel-based system, this would be the size of the smallest protectable file or portion of virtual memory.
GYPSY	A formal program specification language and a verifiable high order language, developed by the University of Texas, designed in conjunction with a complete verification system.

Higher Order Language (HOL)	A computer language which is syntactically and semantically much richer than assembler level languages. HOL's are translated by a compiler or processed by an interpreter to produce executable code or direct execution (interpreter) on most machines, although some machines have hardware interpreters for a specific HOL.
Human Interface Function	A TCB operation that requires human intervention or judgment. Untrusted processes would not be able to invoke them. (See Software Interface Functions.)
Illegitimate Channel	A confinement channel whose presence was not intended. (See legitimate channel, confinement.)
Integrity	The mathematical dual of security that provides protection against unauthorized modification of information (as opposed to unauthorized reading). Whereas higher levels of security restrict the dissemination of information to smaller sets of users, higher levels of integrity restrict the access to commands and capabilities to smaller sets of users.
Integrity Level	The combination of an integrity classification and a set of integrity categories.
Integrity *-property	A rule of the MITRE integrity model that controls reading of information. The integrity level of a subject must be less than or equal to that of an object to read the object.

Inter-Process Communication (IPC)	Communication between two different processes.
Legitimate Channel	A confinement channel whose presence was not intended. (See confinement, covert channel.)
Machine language	The actual sequence of (usually) binary bits which are interpreted by the hardware of a computer to perform a program.
Mandatory Security	The same as non-discretionary security below.
Modula	A Pascal-based higher order language designed to facilitate verification. Used by FACC for KSOS.
Module	A collection of functions which perform operations on and give state information on one component of an abstract machine.
NKSR	Non-Kernel Security-Related software is software that, although related to the security of the overall system, is more convenient to execute in the environment provided by the kernel, as opposed to running as part of the kernel itself. NKSR software usually needs privileges to violate some of the kernel-enforced security rules. (See privileged process)

Non-discretionary Security	The aspect of DoD security policy which deals with security levels. A security level is comprised of a security classification and one or more categories of access restriction. Classifications are totally ordered while categories are partially ordered. To access a piece of information, a user must have a classification greater than or equal to the classification of the information, and at least all the categories of access restriction of the information. The classification and categories of information and users are seldom changed and the accessibility of information by users is easily checked mathematically, without discretion.
Object	The mathematical model abstraction of a repository of information in a computer system.
Pascal	A popular higher level language for which compilers exist on several machines. While Pascal was designed on a general purpose language for ease of writing correct programs, there have been improvements and extensions, especially for system programming, developed for several machines.
Privileged Process	See Trusted Process.
Process	A process consists of a unique address space containing its accessible program code and data, a program location for the currently executing instruction, and periodic access to the processor in order to continue. Unlike a program, which is a static entity, a process is dynamic for it can change the programs in its address space.

Reference Monitor	A concept of control within an abstract machine that limits the access by subjects to objects. A security kernel is an implementation of a reference monitor for a given hardware base.
Secure Path	See Trusted Path.
Security Kernel	A mechanism within a computer system, comprised of hardware and software, that controls the access of users (and processes executing on their behalf) to repositories of information resident in or connected to the system. TCBs have been implemented using security kernels along with trusted processes.
Security Level	The combination of a classification and a set of categories that controls non-discretionary (mandatory) access to information, (i.e., unauthorized (read) access vs. unauthorized modification (write).
Security *-property	The security model rule that prohibits a subject from writing an object of lower security level.
Security Violation	The infringement or breach of a security rule.
Simple Integrity Condition	An integrity rule that controls writing of information. A subject must have an integrity level greater than or equal to that of an object to write the object.
Simple Security Condition	The security model rule that prohibits the access by a subject to information of greater security level than that of the subject.

Software Interface Function	A TCB operation that can be invoked by software, as opposed a person at a terminal. (See Human Interface Function.)
Storage Channel	A channel implemented in software that does not exploit the concurrent execution of two or more processes. A direct storage channel uses the storage associated with a kernel protected information repository (object). An indirect storage channel uses other information associated with kernel objects, such as control information.
Subject	The mathematical model abstraction of a person, process, or other user of information in a computer system.
System High	The highest level in a system, used as in "system high security level" or "system high integrity level".
System Low	The lowest level in a system, used as in "system low security level" or "system low integrity level".
System Utility	Any software, not part of the verified operating system, used to perform various functions that are not directly part of the applications. Examples are compilers, debuggers, editors, and loaders.
Timing Channel	A channel implemented in software that exploits the concurrent execution of two or more processes.

Top-Level Specification (TLS)	A description of the externally observable behavior of a system devoid of implementation detail. The top-level specification of a security kernel precisely defines the behavior of the security kernel observable outside the kernel domain (at the kernel interface). For some verification methodologies, it is an unambiguous description of a finite state machine, written in a machine processable language with a well-defined syntax and semantics. Computer readability of the specifications allows for automation of various phases of verification. It is also called formal specification.
Tranquility	The security model rule that states that the security level of an active object cannot change.
Trusted Computing Base (TCB)	The totality of protection mechanisms for an operating system. It provides both a basic protection environment plus additional user services required for a trustworthy turnkey system. TCBs have been implemented as a security kernel and trusted processes.
Trusted Path	A connection between a user at a terminal and some verified (secure) code that is maintained only by secure code. Use of a secure path assures a user that the intended function will faithfully be presented to the code that executes the function. A secure path is achieved by the user by striking the attention character at a terminal. Also called secure path.

Trusted Process	A process that must be "trusted" to execute as specified if system security is to be maintained. A process is best afforded this "trust" through verification. Trusted processes are sometimes used to execute NKSR software. They are also often endowed with "privileges" to violate kernel-enforced rules.
Verification	The process of mathematically proving that a mechanism behaves in a specified manner. In this context, verification is taken to mean the mathematical proof of correctness of a software system (e.g., the security kernel).
Virtual Space	See Address Space.
UNIX	A general-purpose time-sharing computer system designed to provide a good environment for a user to develop and operate information processing and computation systems. UNIX is a trade/service mark of Western Electric.
Untrusted Process	A process whose incorrect or malicious execution cannot effect system security. (One that has not been subjected to verification.)
*-Property	See Security *-Property, Integrity *-Property.